可下载教学资料

http://www.tup.tsinghua.edu.cn

高等学校教材

计算机应用

计算机网络安全管理

（第2版）

葛秀慧 田浩 金素梅 编著



清华大学出版社

高等学校教材·计算机应用

计算机网络安全管理

（第2版）

葛秀慧　田　浩　金素梅　编著

清华大学出版社

北　京

内 容 简 介

全书共10章，介绍了网络安全的基础知识、加密基础知识、Windows系列操作系统和Linux网络操作系统的安全管理、路由器的安全管理及具体配置策略与方法、电子邮件服务的安全及服务器的安全配置、病毒的基本知识和病毒的查杀、防火墙安全分析及配置实例、电子商务网站的安全及SSL协议配置。本书内容丰富，讲解深入浅出。通过本书的学习可以对网络安全有一个全面而系统的认知，同时可以学会网络安全性工具的使用方法。

本书作为网络管理员和信息安全管理人员的必备手册，既可以作为高等院校信息安全相关专业本科生和专科生的教材，也可供从事相关专业的教学、科研和工程技术人员参考。

图书在版编目（CIP）数据

计算机网络安全管理 / 葛秀慧，田浩，金素梅编著. —2版. —北京：清华大学出版社，2008.5
（高等学校教材 • 计算机应用）
ISBN 978-7-302-17066-2

Ⅰ.计…　Ⅱ.①葛…　②田…　③金…　Ⅲ.计算机网络－安全技术－高等学校－教材
Ⅳ.TP393.08

中国版本图书馆CIP数据核字（2008）第021385号

责任编辑：闫红梅　赵晓宁
责任校对：时翠兰
责任印制：

出版发行：清华大学出版社　　**地　　址**：北京清华大学学研大厦A座
http://www.tup.com.cn　　**邮　　编**：100084
社 总 机：010-62770175　　**邮　　购**：010-62786544
投稿与读者服务：010-62776969，c-service@tup.tsinghua.edu.cn
质量反馈：010-62772015，zhiliang@tup.tsinghua.edu.cn
印 刷 者：
装 订 者：
经　　销：全国新华书店
开　　本：185×260　**印　张**：23　　**字　数**：554千字
版　　次：2008年5月第2版　　**印　次**：2008年5月第1次印刷
印　　数：
定　　价：　　元

编审委员会成员

（按地区排序）

南京邮电学院	朱秀昌	教授
苏州大学	龚声蓉	教授
江苏大学	宋余庆	教授
武汉大学	何炎祥	教授
华中科技大学	刘乐善	教授
中南财经政法大学	刘腾红	教授
华中师范大学	王林平	副教授
	魏开平	副教授
	叶俊民	教授
国防科技大学	赵克佳	教授
	肖　侬	副教授
中南大学	陈松乔	教授
	刘卫国	教授
湖南大学	林亚平	教授
	邹北骥	教授
西安交通大学	沈钧毅	教授
	齐　勇	教授
长安大学	巨永峰	教授
西安石油学院	方　明	教授
西安邮电学院	陈莉君	教授
哈尔滨工业大学	郭茂祖	教授
吉林大学	徐一平	教授
	毕　强	教授
长春工程学院	沙胜贤	教授
山东大学	孟祥旭	教授
	郝兴伟	教授
山东科技大学	郑永果	教授
中山大学	潘小轰	教授
厦门大学	冯少荣	教授
福州大学	林世平	副教授
云南大学	刘惟一	教授
重庆邮电学院	王国胤	教授
西南交通大学	杨　燕	副教授

出版说明

改革开放以来，特别是党的十五大以来，我国教育事业取得了举世瞩目的辉煌成就，高等教育实现了历史性的跨越，已由精英教育阶段进入国际公认的大众化教育阶段。在质量不断提高的基础上，高等教育规模取得如此快速的发展，创造了世界教育发展史上的奇迹。当前，教育工作既面临着千载难逢的良好机遇，同时也面临着前所未有的严峻挑战。社会不断增长的高等教育需求同教育供给特别是优质教育供给不足的矛盾，是现阶段教育发展面临的基本矛盾。

教育部一直十分重视高等教育质量工作。2001 年 8 月，教育部下发了《关于加强高等学校本科教学工作，提高教学质量的若干意见》，提出了十二条加强本科教学工作提高教学质量的措施和意见。2003 年 6 月和 2004 年 2 月，教育部分别下发了《关于启动高等学校教学质量与教学改革工程精品课程建设工作的通知》和《教育部实施精品课程建设提高高校教学质量和人才培养质量》文件，指出“高等学校教学质量和教学改革工程”是教育部正在制定的《2003—2007 年教育振兴行动计划》的重要组成部分，精品课程建设是“质量工程”的重要内容之一。教育部计划用五年时间（2003—2007 年）建设 1500 门国家级精品课程，利用现代化的教育信息技术手段将精品课程的相关内容上网并免费开放，以实现优质教学资源共享，提高高等学校教学质量和人才培养质量。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作，提高教学质量的若干意见》精神，紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”，在有关专家、教授的倡议和有关部门的大力支持下，我们组织并成立了“清华大学出版社教材编审委员会”（以下简称“编委会”），旨在配合教育部制定精品课程教材的出版规划，讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师，其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求，“编委会”一致认为，精品课程的建设工作从开始就要坚持高标准、严要求，处于一个比较高的起点上；精品课程教材应该能够反映各高校教学改革与课程建设的需要，要有特色风格、有创新性（新体系、新内容、新手段、新思路，教材的内容体系有较高的科学创新、技术创新和理念创新的含量）、先进性（对原有的学科体系有实质性的改革和发展、顺应并符合新世纪教学发展的规律、代表并引领课程发展的趋势和方向）、示范性（教材所体现的课程体系具有较广泛的辐射性和示范性）

和一定的前瞻性。教材由个人申报或各校推荐（通过所在高校的“编委会”成员推荐），经“编委会”认真评审，最后由清华大学出版社审定出版。

目前，针对计算机类和电子信息类相关专业成立了两个“编委会”，即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。首批推出的特色精品教材包括：

（1）高等学校教材·计算机应用——高等学校各类专业，特别是非计算机专业的计算机应用类教材。

（2）高等学校教材·计算机科学与技术——高等学校计算机相关专业的教材。

（3）高等学校教材·电子信息——高等学校电子信息相关专业的教材。

（4）高等学校教材·软件工程——高等学校软件工程相关专业的教材。

（5）高等学校教材·信息管理与信息系统。

（6）高等学校教材·财经管理与计算机应用。

清华大学出版社经过 20 多年的努力，在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌，为我国的高等教育事业做出了重要贡献。清华版教材形成了技术准确、内容严谨的独特风格，这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会
E-mail: dingl@tup.tsinghua.edu.cn

前言

在信息时代，信息安全越来越重要。现在大部分信息都是通过网络来传播，网络安全成为21世纪世界十大热门课题之一。网络安全在IT业内可分为网络安全硬件、网络安全软件和网络安全服务。网络安全硬件包括防火墙和VPN、独立的VPN、入侵检测系统、认证令牌和卡、生物识别系统、加密机和芯片；网络安全软件包括安全内容管理、防火墙/VPN、入侵检测系统、安全3A、加密，其中安全内容管理还有防病毒、网络控制和邮件扫描，安全3A包括授权、认证和管理；网络安全服务包括顾问咨询、设计实施、支持维护、教育培训和安全管理。目前，随着因特网的日益普及，网络安全正在成为人们关注的焦点。而要保证网络安全就必须对网络进行安全的管理。

全书分为10章，第1章讲述了网络安全的基础知识。第2章详细地讲述了加密技术，加密技术作为一种主动的防卫手段，是网络安全最有效的技术之一。一个加密网络，不但可以防止非授权用户的搭线窃听和入网，而且也是对付恶意软件的有效方法。第3～第5章分别对当前流行的操作系统Windows NT、Windows Server 2003和Linux的安全管理进行了详细的分析和阐述。信息安全对今天的网络系统来说，是一个非常重要又非常严重的问题，它涉及从硬件到软件、从单机到网络的各个方面的安全性机制。而网络操作系统的安全性是整个网络系统安全体系中的基础环节，所以对网络操作系统的安全配置是极其重要的。第6章详细地介绍了路由器的安全管理，因为在目前的网络体系中，路由器是多种网络互联的重要设备，路由器一般位于防火墙之外，是边界网络的前沿，路由器的安全管理成为了第一道防线，所以路由器的安全越来越受到重视。第7章对电子邮件服务的安全以及客户端和邮件服务器的安全配置进行了详尽的介绍。如今病毒的总数以每月上百个的速度增加，如蠕虫病毒、CIH病毒、BO黑客程序、宏病毒、求职信病毒及以电子邮件传染的邮件病毒，都借助于网络——这个世界性的传播途径而具备了更强的攻击力，所以第8章对病毒的基本知识和病毒的查杀做了专门的讲解。防火墙是一种被动的防御技术，是一类防范措施的总称，是目前在网络安全技术中使用最多、最广泛的一种安全技术。第9章对防火墙进行了分析，并给出了配置实例。如何建立一个安全、快捷的电子商务应用环境，对信息提供足够的保护，已经成为商家和用户都十分关心的话题。所以要开展电子商务，就必须充分了解电子商务中应该注意的安全问题。第10章对电子商务网站的安全配置及SSL协议进行了详细的分析。

通过本书的学习可以对网络安全有一个全面而系统的认知，同时可以学会网络安全性工具的使用方法。本书适用于网络管理员和信息安全管理人员，既可以作为高等院校信息安全相关专业本科生和专科生的教材，也可供从事相关专业的教学、科研和工程技术人员参考。

本书由葛秀慧、田浩和金素梅等编著。由于编者水平有限、经验不足，缺点和错误在所难免，希望读者多提宝贵意见，诚望专家和广大读者不吝赐教，批评指正。

编　者

2008 年 3 月

目 录

第1章

网络安全管理基础

在信息时代，信息安全问题越来越重要。现在，大部分信息都是通过网络进行传播的，网络安全成为21世纪世界十大热门课题之一。网络安全在IT业内可分为网络安全硬件、网络安全软件和网络安全服务。其中，网络硬件包括防火墙和VPN、独立的VPN、入侵检测系统、认证令牌环卡、生物识别系统、加密机和芯片。网络安全软件包括安全内容管理、防火墙和VPN、入侵检测系统、安全3A、加密等。其中安全内容管理还包括防病毒、网络控制和邮件扫描，安全3A包括授权、认证和管理。网络安全服务包括顾问咨询、设计实施、支持维护、教育培训和安全管理。随着因特网的日益普及，网络安全正在成为一个受人关注的焦点。而要保证网络安全就必须对网络进行安全管理。下面先了解一下网络体系结构的相关知识，以及在相应模型中的安全问题，再对网络安全进行详细的分析及讨论解决网络安全管理问题的关键技术。

1.1　网络体系结构概述

众所周知，一个计算机网络有许多互相连接的节点，在这些节点之间要不断地进行数据交换。要做到有序地交换数据，每个节点就必须遵守一些事先约定好的规则，这些规则明确规定了所交换数据的格式及相关的同步问题。这些为进行网络数据交换而建立的规则、标准或约定就称为网络协议。一个网络协议主要由以下三个要素组成。

- 语法：数据与控制信息的结构或格式。
- 语义：需要发出何种控制信息、完成何种协议及做出何种应答。
- 同步：事件实现顺序的详细说明。

由此可见，网络协议是计算机网络不可缺少的部分。很多经验和实践表明，对于非常复杂的计算机网络协议，为了减少网络设计的复杂性，大多数网络都按层（layer）或级（level）的方式来进行组织。不同的网络，其层的数量、名字、内容和功能都不尽相同。这样分层的好处在于：每一层都实现相对的独立功能，因此就能将一个难以处理的复杂问题分解为若干个较容易处理的问题。

计算机网络的各层及协议的集合称为网络的体系结构（network architecture）。换言之，计算机网络的体系结构是使这个计算机网络及其部件所应该完成的功能的精确定义。需要

强调的是，这些功能究竟由何种硬件或软件完成，则是一个遵循这种体系结构的实现的问题。可见，体系结构是抽象的，是存在于纸上的，而对它的实现是具体的，是运行在计算机软件和硬件之上的。常见的网络层次结构如图 1-1 所示。

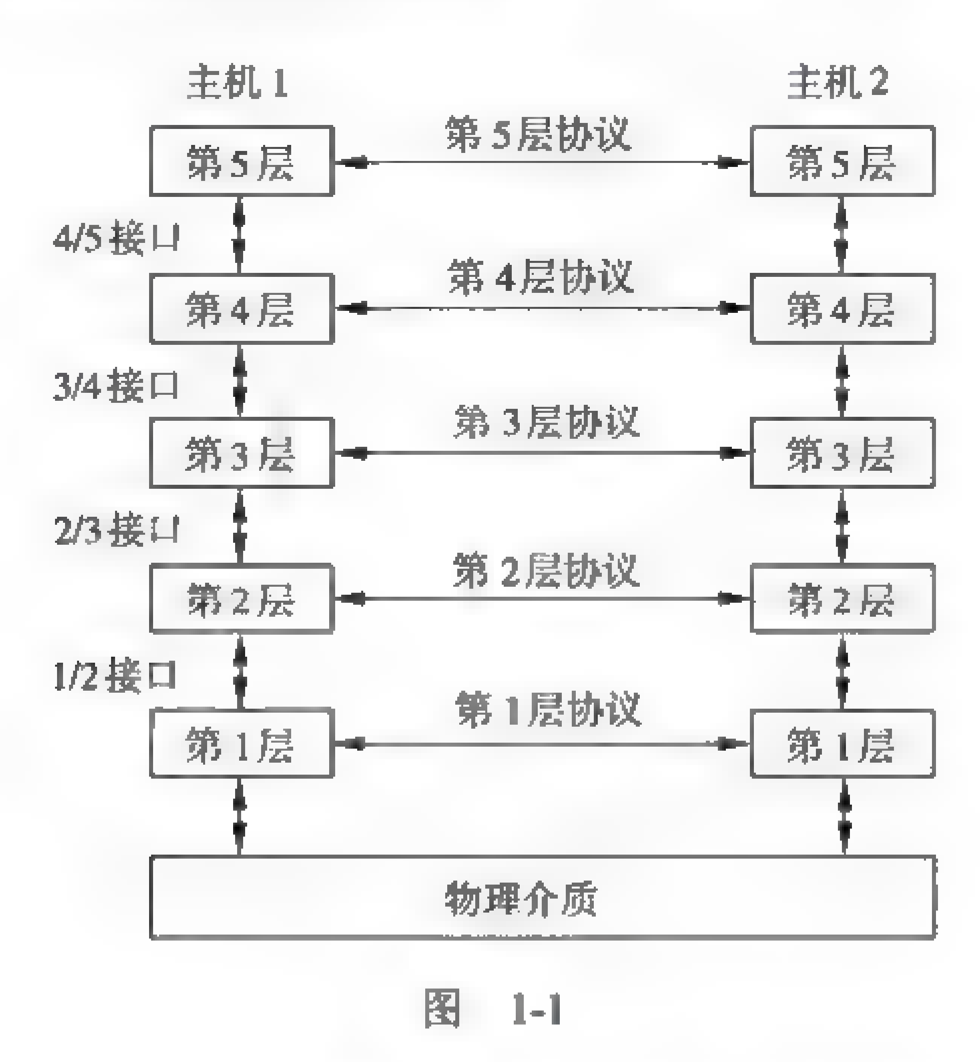


图 1-1

1.2 网络体系结构的参考模型

网络体系结构的参考模型主要有两种：OSI 模型和 TCP/IP 模型。

1.2.1 OSI 参考模型

现代计算机网络的设计，是按高度结构化方式进行的。为减少协议设计的复杂性，大多数网络都按层或级的方式来组织，每一层都建立在它的下层之上。不同的网络，其层的数量，各层的名字、内容和功能都不尽相同。然而，在所有的网络中，每一层的目的都是向它的上一层提供服务的，而把这种服务是如何实现的细节对上层加以屏蔽。

最著名的网络体系结构是国际标准化组织 ISO 的开放系统互连（Open System Interconnection，OSI）参考模型，即通常所提的 OSI 模型。OSI 模型有 7 层，其分层原则如下：

- 根据功能的需要分层。
- 每一层应当实现一个定义明确的功能。
- 每一层功能的选择应当有利于制定国际标准化协议。
- 各层界面的选择应当尽量减少通过接口的信息量。
- 层数应足够多，以避免不同的功能混杂在同一层中。但也不能过多，否则体系结构会过于庞大。

OSI 参考模型由低到高依次是物理层、数据链路层、网络层、传输层、会话层、表示层和应用层，其体系结构如图 1-2 所示。

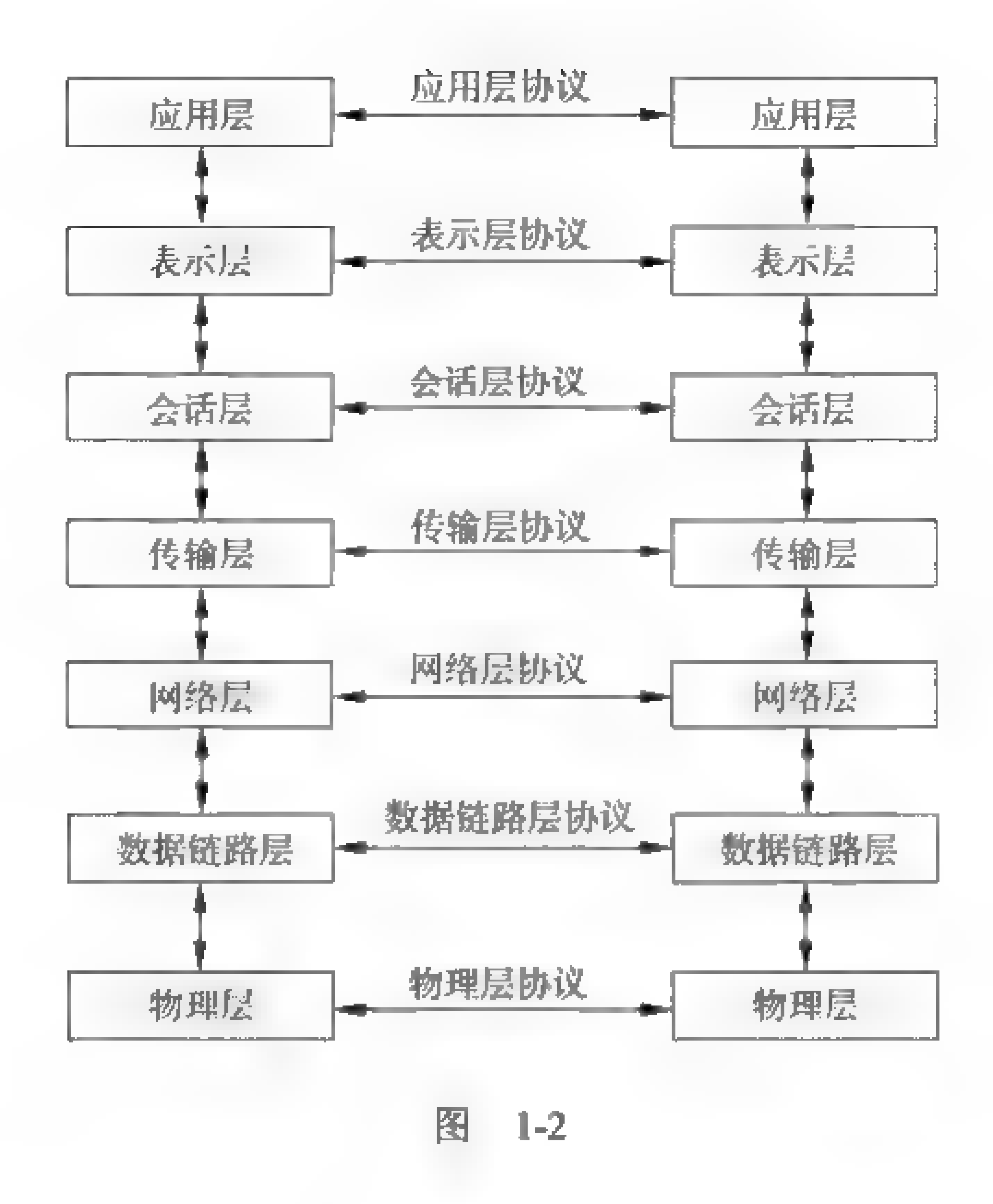


图　1-2

1.2.2　TCP/IP 协议结构体系

OSI 参考模型的建立是计算机网络技术发展的一个里程碑，它为网络的标准化提供了一致的框架和前景。但由于 OSI 参考模型的庞大，因此在建立网络时，并没有完全依赖 OSI 参考模型。事实上，基于 TCP/IP 协议的 Internet 网络有着自己的网络体系结构——TCP/IP 网络体系结构。这种体系结构，目前已经成为事实上的网络标准。

TCP/IP 协议体系结构与 OSI 参考模型类似，也为分层体系结构，但比 OSI 参考模型的层数要少，一般为 4 层结构，从低到高依次为网络接口层、网络层、传输层和应用层，如图 1-3 所示。

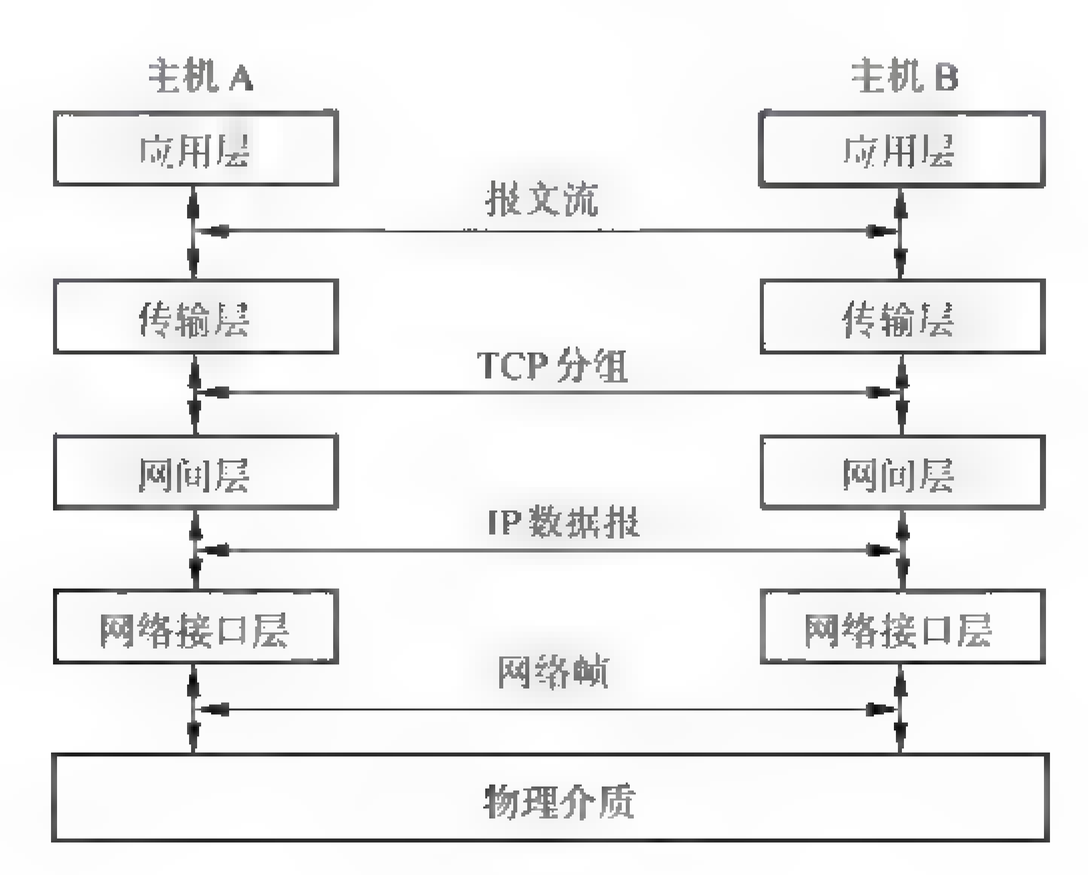


图　1-3

1．网络接口层

网络接口层在 TCP/IP 协议结构的最底层。该层中的协议提供了一种数据传送的方法，使得系统可以通过直接的物理连接的网络，将数据传送到其他设备，并定义了如何利用网络来传送 IP 数据报。TCP/IP 网络接口层一般包括 OSI 参考模型的物理层和数据链路层的全部功能，因此这一层的协议很多，包括各种局域网、广域网的各种物理网络的标准。

2．网络层

网络层在网络接口的上一层。网络层协议 IP 是 TCP/IP 的核心协议，也是网络层中最重要的协议。IP 可提供基本的分组传输服务，这是构造 TCP/IP 的基础。网络层上、下层中的所有协议都使用 IP 协议传送数据：所有的 TCP/IP 数据，无论是进来的还是出去的，都流经 IP，并与它的最终目的地无关。另外，网络层还有地址转换协议（ARP）和网间控制报文协议（ICMP）两个协议，其中 ICMP 协议具有测试网络链路和检测网络故障的功能，是 IP 协议不可分割的一部分。

3．传输层

传输层在网络层的上一层，又称主机到主机传输层。传输层有传输控制协议（TCP）和用户数据报协议（UDP）两个重要的协议，用以提供端到端的数据传输服务，即从一个应用程序到另一个应用程序之间的信息传递。TCP 利用端到端的错误检测与纠正功能，提供可靠的数据传输服务。而 UDP 则提供低开销、无链接的数据报传输服务。

4．应用层

TCP/IP 协议体系结构的顶层是协议最多的一层。应用层的协议大多数都为用户提供直接的服务，而且还在不断地增加新的服务。

常见的应用层协议有：

- Telnet 网络终端协议。
- FTP 文件传输协议。
- SMTP 简单邮件传输协议。
- POP 邮件接收协议。
- HTTP 超文本传输协议。
- DNS 域名服务等。

1.3 系统安全结构

网络系统的安全涉及平台的各个方面。按照网络 OSI 的 7 层模型，网络安全贯穿于整个 7 层模型。针对网络系统实际运行的 TCP/IP 协议，网络安全贯穿于信息系统的 4 个层次。网络的安全体系层次模型如表 1-1 所示。

表 1-1　网络的安全体系层次模型

应用系统	应用系统安全	应用系统	应用系统安全
应用平台	应用平台安全	链路层	链路安全
会话层	会话安全	物理层	物理层安全
网络层	安全路由/访问机制		

1．物理层

物理层信息安全，主要防止物理通路的损坏、物理通路的窃听及对物理通路的攻击（干扰等）。

2．数据链路层

数据链路层的网络安全需要保证通过网络链路传送的数据不被窃听，主要采用划分 VLAN（局域网）、加密通信（远程网）等手段。

3．网络层

网络层的安全需要保证网络只给授权的客户使用授权的服务，保证网络路由正确，避免被拦截或监听。

4．操作系统

操作系统安全要求保证客户资料、操作系统访问控制的安全，同时能够对该操作系统上的应用进行审计。

5．应用平台

应用平台指建立在网络系统之上的应用软件服务，如数据库服务器、电子邮件服务器和 Web 服务器等。由于应用平台的系统非常复杂，通常采用多种技术（如 SSL 等）来增强应用平台的安全性。

6．应用系统

应用系统完成网络系统的最终目的——为用户服务。应用系统的安全与系统设计和实现关系密切，应用系统使用应用平台提供的安全服务来保证基本安全，如通信内容安全，通信双方的认证、审计等手段。

1.4　TCP/IP 层次安全

TCP/IP 的层次不同，提供的安全性也不同，例如，在网络层提供虚拟私用网络，在传输层提供安全套接字服务。下面将分别介绍 TCP/IP 不同层次的安全性和提高各层安全性的方法。

1.4.1 网络层的安全性

国际上正在对网络层（Internet 层）的安全协议进行标准化。如“安全协议 3 号（SP3）”、“网络层安全协议（NLSP）”、“集成化 NLSP（I-NLSP）”、SwIPe 和 IPSP 等安全协议，这些安全协议用的都是 IP 封装技术。其本质是：纯文本的包被加密、封装在外层的 IP 报头里，用来对加密的包进行因特网上的路由选择。到达另一端时，外层的 IP 报头被拆开，报文被解密，然后送到收报地点。

网络层安全性的主要优点是它的透明性，即安全服务的提供，不需要应用程序、其他通信层次和网络部件做任何改动。它的主要缺点是网络层一般对属于不同进程和相应条例的包不加以区别。对所有发往同一地址的包，它将按照同样的加密密钥和访问控制策略来处理。这可能导致提供不了所需的功能，也会导致性能下降。

网络层是非常适合提供基于主机对主机的安全服务的。相应的安全协议可以用来在因特网上建立安全的 IP 通道和虚拟私有网。例如，利用它对 IP 包的加密和解密功能，可以简捷地强化防火墙系统的防卫能力。

1.4.2 传输层的安全性

在因特网中提供安全服务的首先想法便是强化它的 IPC（Internet Protocol Control）界面，如 BSD Sockets 等，具体做法包括双端实体的认证、数据加密密钥的交换等。Netscape 通信公司遵循了这个思路，制定了建立在可靠的传输服务（如 TCP/IP 所提供）基础上的安全套接（层）协议（SSL）。SSL 版本 3（SSL v3）于 1995 年 12 月制定，主要包含以下两个协议。

1. SSL 记录协议

该协议涉及应用程序提供的信息的分段、压缩、数据认证和加密。SSL v3 提供对数据认证用的 MD5 和 SHA 及数据加密用的 R4 和 DES 等的支持，用来对数据进行认证和加密的密钥可以通过 SSL 的握手协议来协商。

2. SSL 握手协议

用来交换版本号、加密算法、（相互）身份认证并交换密钥。SSL v3 提供对 Diffie-Hellman 密钥交换算法、基于 RSA 的密钥交换机制和另一种实现在 Fortezza chip 上的密钥交换机制的支持。

1.4.3 应用层的安全性

网络层（或传输层）的安全协议允许为主机（或进程）之间的数据通道增加安全属性。本质上，这意味着真正的（或许再加上机密的）数据通道还是建立在主机（或进程）之间，但却不可能区分在同一通道上传输的一个具体文件的安全性要求。例如，如果一个主机与

另一个主机之间建立起一条安全的 IP 通道，那么所有在这条通道上传输的 IP 包都要自动地被加密。同样，如果一个进程和另一个进程之间通过传输层安全协议建立起了一条安全的数据通道，那么两个进程间传输的所有消息都要自动地被加密。

如果确实想要区分一个具体文件的不同的安全性要求，那就必须借助于应用层的安全性。提供应用层的安全服务实际上是最灵活的处理单个文件安全性的手段。例如一个电子邮件系统可能需要对要发出的信件的个别段落实施数据签名，较低层的协议提供的安全功能一般不会知道任何要发出的信件的段落结构，从而不可能知道应该对哪一部分进行签名。只有应用层是唯一能够提供这种安全服务的层次。

目前都使用 PKI（公钥基础结构）进行认证和密钥分配。PEM PKI 是按层次组织的，由下述三个层次构成：

- 顶层为网络层安全政策登记机构（IPRA）。
- 第二层为安全政策证书颁发机构（PCA）。
- 底层为证书颁发机构（CA）。

1.5　TCP/IP 的服务安全

对 TCP/IP 协议的服务很多，人们比较熟悉的有 WWW 服务、FTP 服务和电子邮件服务，不太熟悉的有 TFTP 服务、NFS 服务和 Finger 服务等。这些服务都存在不同程度的安全缺陷，当用户用防火墙保护站点时，就需要考虑应该提供哪些服务，要禁止哪些服务，在这里只对一些服务进行介绍。

1.5.1　WWW 服务

WWW 服务相对于其他服务出现比较晚，它基于超文本传输协议（HTTP），是人们最常使用的 Internet 服务。随着 Netscape 公司推出安全套接字层，WWW 服务器和浏览器的安全性得到大大的提高，现在人们已经把这种技术应用于电子商务（E-business），例如在许多国家，人们可以在因特网上买卖股票和使用信用卡购物。既然把 WWW 服务说得那么安全，那么它是否存在安全问题呢？安全套接字层确实保证了 WWW 服务的安全，但它主要解决了数据包被窃听和劫持的问题。除此之外，WWW 服务还有其他问题，如 WWW 服务使用的 CGI 程序、服务器端附件（Server Side Include，SSI）和 Java Applet 小程序等。

1.5.2　电子邮件服务

电子邮件服务给人们提供了一种方便和快捷的服务，现在大部分人都有一个或多个 E-mail 地址。目前，任何一个大型网站上都有免费或收费电子邮件的申请。但是电子邮件附件中的 Word 文件或其他文件中有可能会带有病毒。还有电子邮件炸弹也是一个令人头疼的问题。收到了一大堆垃圾邮件，直到邮件箱被塞满，也会给用户带来不便。关于电子邮件的安全将在第 6 章中讲述，希望对读者了解电子邮件安全起到一定的作用。

1.5.3 FTP 服务和 TFTP 服务

FTP 服务和 TFTP 服务都是用于传输文件的，但用的场合不同，安全程度也不同。TFTP 服务用于局域网，在无盘工作站启动时用于传输系统文件，安全性极差，常被人用来窃取密码文件（/etc/passwd），因为它不带有任何安全认证。FTP 服务对于局域网和广域网都可以，可以用来下载任何类型的文件。网上有许多匿名 FTP 服务站点，上面有许多免费软件、图片和游戏，匿名 FTP 是人们常使用的一种服务方式。FTP 服务的安全性要好一些，起码它需要用户输入用户名和口令。当然，匿名 FTP 服务就像匿名 WWW 服务一样是不需要口令的，但用户权力会受到严格的限制。匿名 FTP 存在一定的安全隐患，因为有些匿名 FTP 站点提供可写区为用户所使用，这样用户可以上传一些软件到站点上。但这些可写区常被一些人作为地下仓库，存放一些盗版软件和黄色图片，这会浪费用户的磁盘空间、网络带宽等系统资源，可能会造成“拒绝服务”攻击。匿名 FTP 服务的安全很大程度上决定于一个系统管理员的水平。一个低水平的系统管理员很可能会错误配置权限，从而被黑客利用破坏整个系统。

1.5.4 Finger 服务

Finger 服务用于查询用户的信息，包括网上成员的真实姓名、用户名、最近的登录时间和地点等，也可以用来显示当前登录在机器上的所有用户名。这对于入侵者来说是无价之宝，因为它能告诉他在本机上的有效登录名，然后入侵者就可以注意其活动。

1.5.5 其他服务

除了上面提到的 Finger 和 TFTP，还有 X-Window 服务，基于 RPC 的 NFS 服务，BSD UNIX 的以 r 开头的服务，如 rlogin、rsh 和 rexec 等。这些服务在设计上安全性很差，一般只在内部网使用。如果有防火墙，应把这些服务限制在内部网中。

1.6 个人网络安全

关于个人网络的安全是很重要的，如果网上免费邮箱被炸，上网账号被偷用等，对用户就造成了损失。下面简单介绍一下应该注意的问题。

- 邮箱中标题不明的邮件不能随便打开。
- 在聊天室或 BBS 上不公开自己的 IP、邮件地址等个人隐私。
- 要经常更换自己计算机的密码。另外，不要在外人面前输入密码，密码的长度要足够长。
- 一般不要让计算机记住密码，以免别人使用你的计算机使自己的机密外泄。
- 不要在自己的计算机上运行不明的程序，这些可能是黑客程序。

1.7　局域网的安全

目前的局域网基本上都采用以广播为技术基础的以太网，任何两个节点之间的通信数据包，不仅为这两个节点的网卡所接收，也同时为处在同一以太网上的任何一个节点的网卡所截取。因此，黑客只要接入以太网上的任一节点进行侦听，就可以捕获发生在这个以太网上的所有数据包，对其进行解包分析，从而窃取关键信息，这就是以太网所固有的安全隐患。

局域网安全方法有如下几种。

1.7.1　网络分段

网络分段是保证安全的一项重要措施，就是将非法用户与网络资源相互隔离，从而达到限制用户非法访问的目的。

网络分段可分为物理分段和逻辑分段两种方式。

1．物理分段

物理分段通常是指将网络从物理层和数据链路层（ISO/OSI 模型中的第 1 层和第 2 层）上分为若干网段，各网段相互之间无法进行直接通信。目前，许多交换机都有一定的访问控制能力，可实现对网络的物理分段。

2．逻辑分段

逻辑分段是指将整个系统在网络层（ISO/OSI 模型中的第 3 层）上进行分段。例如，对于 TCP/IP 网络，可把网络分成若干 IP 子网，各子网间必须通过路由器、路由交换机、网关或防火墙等设备进行连接，利用这些中间设备（含软件、硬件）的安全机制来控制各子网间的访问。在实际应用过程中，通常采取物理分段与逻辑分段相结合的方法来实现对网络系统的安全性控制。

1.7.2　以交换式集线器代替共享式集线器

对局域网的中心交换机进行网络分段后，以太网侦听的危险仍然存在。这是因为网络最终用户的接入往往是通过分支集线器而不是中心交换机，而使用最广泛的分支集线器通常是共享式集线器。这样，当用户与主机进行数据通信时，两台机器之间的数据包（为单播包 Unicast Packet）还是会被同一台集线器上的其他用户所侦听。一种很危险的情况是，用户远程登录到一台主机上，由于 TELNET 程序本身缺乏加密功能，用户所输入的每一个字符（包括用户名、密码等重要信息）都将被明文发送，这就给黑客提供了机会。

因此，应该以交换式集线器代替共享式集线器，使单播包仅在两个节点之间传送，从而防止非法侦听。

1.7.3 虚拟专网

虚拟专网技术主要基于近年发展的局域网交换技术（ATM 和以太网交换）。交换技术将传统的基于广播的局域网技术发展为面向连接的技术。因此，网管系统有能力限制局域网通信的范围而无须通过开销很大的路由器。

以太网从本质上是广播机制，但应用了交换器和 VLAN 技术后，实际上转变为点对点通信。除非设置了监听口，信息交换才不会存在监听和插入（改变）问题，所以运行虚拟网技术带来的网络安全的好处是显而易见的。

1.8 广域网的安全

由于广域网大多采用公网来进行数据传输，因此信息在广域网上传输时被截取和利用的可能性就比局域网要大得多。

广域网安全解决办法主要依靠防火墙技术、入侵检测技术和网络防病毒技术。在实际的广域网安全设计中，往往采取上述三种技术（防火墙、入侵检测和网络防病毒）相结合的方法。广域网一般采用以下安全解决办法。

1.8.1 加密技术

加密型网络安全技术的基本思想是不依赖于网络中数据通道的安全性来实现网络系统的安全，而是通过对网络数据的加密来保障网络的安全可靠性。数据加密技术可以分为三类：对称型加密、不对称型加密和不可逆加密。

其中不可逆加密算法不存在密钥保管和分发问题，适用于分布式网络系统，但是其加密计算量相当可观，所以通常在数据量有限的情形下使用。计算机系统中的口令就是利用不可逆加密算法加密的。将在第 2 章中详细讲述加密技术知识。

1.8.2 VPN 技术

虚拟专网（Virtual Private Network，VPN）技术的核心是采用隧道技术，将企业专网的数据加密封装后，透过虚拟的公网隧道进行传输，从而防止敏感数据的被窃。VPN 可以在 Internet、服务提供商的 IP、帧中继或 ATM 网上建立。企业通过公网建立 VPN，就如同通过自己的专用网建立内部网一样，享有较高的安全性、优先性、可靠性和可管理性，同时还为移动计算提供了可能。因此，VPN 技术一经推出就深得人心，是一种很好的安全技术。

1.8.3 身份认证技术

对于从外部拨号访问总部内部网的用户，由于使用公共电话网进行数据传输所带来的

风险，必须更加严格控制其安全性。一种常见的做法是采用身份认证技术，对拨号用户的身份进行验证并记录完备的登录日志。较常用的身份认证技术，有 Cisco 公司提出的 TACACS+及业界标准的 RADIUS 等。

1.9　网络安全威胁

网络安全威胁是指有可能访问资源并造成破坏的某个人、某个地方或某个事物。威胁的类型很多，有自然的和物理的（火灾、地震），无意的（不知情的顾客或员工）和故意的（攻击者、恐怖分子和工业间谍等）。下面介绍网络威胁的几个重要概念。

1. 安全漏洞

安全漏洞是指资源容易遭受攻击的位置，它可以被视为一个弱点。安全漏洞通常按表 1-2 所示的方法进行分类。安全漏洞类型有物理的和自然的、硬件和软件的、媒介的、通信的（未加密的协议）和人为的等。

表 1-2　计算环境中的漏洞

安全漏洞类型	示　例	安全漏洞类型	示　例
物理的	未锁门窗	媒介	电干扰
自然的	灭火系统失灵	通信	未加密协议
硬件和软件	防病毒软件过期	人为	不可靠的技术支持

2. 乘虚攻击

一种威胁可以通过利用环境中的安全漏洞访问到计算机中的资源产生。这种类型的攻击也称为乘虚攻击。乘虚攻击资源的方法有许多种。

1）利用技术漏洞型攻击

- 强力攻击。
- 缓冲区溢出。
- 错误配置。
- 重放攻击。
- 会话劫持。

2）信息收集

- 地址识别。
- 操作系统识别。
- 端口扫描。
- 服务和应用程序探测。
- 漏洞扫描。
- 响应分析。
- 用户枚举。
- 文档研磨。

- 无线泄露。
- 社会工程。

3）拒绝服务

- 物理损坏。
- 资源删除。
- 资源修改。
- 资源饱和。

另外，来自网络的威胁还有如下几方面。

- 操作系统的安全性：许多操作系统均存在网络安全漏洞。
- 防火墙的安全性：防火墙产品是否设置错误等。
- 来自内部网用户的安全威胁。
- 缺乏有效的手段监视、评估网络系统的安全性。
- 采用的TCP/IP协议族软件，本身缺乏安全性。
- 未能对来自Internet的电子邮件挟带的病毒进行有效控制。
- 应用服务的安全：应用服务系统在访问控制及安全通信设置错误等。

1.10 网络系统安全应具备的功能

网络系统的安全体系应包含如下几方面。

- 访问控制：通过对特定网段、服务建立的访问控制体系，将绝大多数攻击阻止在到达攻击目标之前。
- 检查安全漏洞：通过对安全漏洞的周期检查，即使攻击可到达攻击目标，也可使绝大多数攻击无效。
- 攻击监控：通过对特定网段、服务建立的攻击监控体系，可实时检测出绝大多数攻击，并采取相应的行动（如断开网络连接、记录攻击过程和跟踪攻击源等）。
- 加密通信：主动的加密通信，可使攻击者不能了解、修改敏感信息。
- 认证：良好的认证体系可防止攻击者假冒合法用户。
- 备份和恢复：良好的备份和恢复机制，可在攻击造成损失时，尽快地恢复数据和系统服务。
- 多层防御：攻击者在突破第一道防线后，延缓或阻断其到达攻击目标。
- 隐藏内部信息：使攻击者不能了解系统内的基本情况。
- 设立安全监控中心：为信息系统提供安全体系管理、监控、保护及紧急情况服务。

1.11 网络安全的主要攻击形式

在网络安全中常用的攻击形式有信息收集、利用技术漏洞型攻击、会话劫持、防止DNS毒化、URL字符串攻击、攻击安全账户管理器、文件缓冲区溢出、拒绝服务、攻击后门攻击和恶意代码等。

1.11.1　信息收集

攻击者总是要挖空心思找到要攻击环境的信息。防范信息收集的关键技术就是限制外部对资源进行未经授权的访问。经常使用的方法有：

- 确保网络上只有那些已标识的特定设备能够建立远程访问连接。
- 在通过外部防火墙直接连接 Internet 的计算机上关闭 TCP/IP 上的 NetBIOS，包括端口 135、137、139 和 445。这样做能使外部人员更难利用标准联网手段连接到服务器。
- 在面向 Internet 的网络适配器和过滤流向某一网站通信的防火墙上仅启用端口 80 和 443，这样做可以消除大多数基于端口的侦测攻击。
- 审查公共网站上的信息以确保该站点上使用的电子邮件地址不是管理账户。
- 管理放在 Web 站点的源代码中的内容类型，以防止攻击者审阅该代码（该技术有时被称为源代码筛选）来获取宝贵的信息。
- 审查为一般公众提供的信息有没有自己的 IP 地址和域名注册信息。确保攻击者无法通过 DNS 查询参考网络或哄骗 DNS 执行完整的区域复制，因为通过转储 DNS 中的所有记录，攻击者可以清楚地发现最易于攻击的计算机。为了防止 DNS 查询，可以通过利用通知选项和仅允许到授权服务器的区域复制，为 Windows 2000 DNS 服务器分配权限。另一个办法是实施一个只读 DNS，并部署更新它的策略和步骤。

除了防范信息收集的常用 6 种方法外，还应该了解攻击者所使用的信息收集方法。在信息收集中，攻击者最常使用的方法就是扫描方法，表 1-3 中列出了一般攻击者使用的扫描方法及其应用。

表 1-3　扫描的方法及其应用

扫描方法	工作方式	应用
Internet 控制消息协议（ICMP）回显或 ping	将 ICMP 端口 0 数据包发送给接收系统，如果系统允许响应 ICMP 回显，它将给正在扫描的系统发送一个 ICMP 回复，表明系统正在工作并在监听网络通信	Ping 扫描用于识别网络上正在监听的主机，它不能识别 ICMP 之外的监听端口或协议。许多安全过滤设备都会阻止 ICMP 回显请求，因此可防止 ping 信号通过网络周边
TCP 连接或三方握手	利用标准三方握手方式验证到监听 TCP 端口的连接	通不过 TCP 过滤安全设备（比如防火墙或数据包过滤路由器），则会很好
TCP 哄骗连接请求（SYN）	利用三方握手的前两个步骤，正在扫描的系统会发送一个带有上一步重置（RST）标志的数据包，而不是状态确认（ACK），因而不会建立一个完整的连接	由于连接从未建立，因此被安全设备检测出或过滤掉的可能性更小，在某种程度上比 TCP 连接扫描慢
TCP Finish（FIN）	除 FIN 标志外，所有标志均被关闭；监听端口上收到的这类数据包通常不会发出响应，反而非监听端口会发送 RST 数据包；不响应的端口是那些正在监听的端口	可能绕过仅监听 SYN 数据包的系统或安全设备，与 TCP SYN 扫描结果类似。也许不能从基于 Windows 的系统上获得准确的结果，从而难以确定这些系统上已打开的端口
碎片数据包	使用以前的一种扫描技术，将 TCP 数据包分成碎片以在目标位置重新组装	有些安全设备（包括入侵检测系统）在重新组装这些数据包流时可能遇到困难。有时能够绕过过滤设备，甚至导致它们崩溃

续表

扫描方法	工作方式	应用
Ident 检索	建立 TCP 连接(三方握手)之后发出 Ident 请求，以确定哪一个账户与监听端口进程相关联	此类扫描不能识别监听端口，但能够识别账户及其相关服务
文件传输协议（FTP）代理扫描	RFC for FTP 设计了代理类型的服务，可使用户与 FTP 服务器建立连接并请求 FTP 服务器启动面向任何其他系统的文件传输。FTP 代理扫描利用这种设计缺陷来代理与其他系统的端口连接请求	可能在扫描隐藏在防火墙背后的系统时有用。能够发现允许这种扫描的系统这一点本身就是一个漏洞，因为它会向安全策略或安全设备不允许的位置传输通信
UDP	UDP 是一种无连接协议，这意味着发送系统不需要从目标系统得到响应。执行 UDP 扫描的系统只会从非监听端口收到响应	UDP 端口通常不会被安全设备过滤掉，或仅被有限过滤。UDP 服务 DNS 和简单网络管理协议（SNMP）没有得到安全的实现，而且通常被允许通过网络周边，即通过通信子网进入资源子网。可能显示打开的大多数端口
OS 检测	OS 检测的执行方式有多种，但最准确的方式通常是将从设备收到的 TCP 响应与已知系统类型列表相比较。用于确定主机信息的一些组件包括 TTL、TCP 序列号、分段、FIN 和 ACK 响应、未定义的标志响应、窗口大小、ICMP 响应及多个 TCP 选项	OS 检测扫描通常会绕过许多过滤设备，但代理防火墙除外，因为防火墙就是实际发出响应的设备。可能会返回多个 OS 类型，而且结果可能不准确。防火墙或路由器通常会拒绝基于 ICMP 的 OS 检测扫描

1.11.2　利用技术漏洞型攻击

攻击者会企图利用环境中的技术漏洞，以获取对系统的访问权限并提升其权限。有许多可以实现的方法。在本节中，列出了一些主要方法，并介绍了相应的防范措施。

1. 会话劫持

攻击者使用会话劫持工具来中断、中止或窃取正在进行的会话。这些攻击类型重点针对基于会话的应用程序，许多会话劫持工具能同时查看多个会话。防范会话劫持保护体系结构的最佳办法就是进行加密。

2. URL 字符串攻击

攻击者开始将重点放在遍历端口 80 的攻击上。其中一种攻击形式就是创建一个利用 Unicode Translation Format-8（UTF-8）编码的正斜杠或反斜杠（/或\）版本的 URL 字符串，例如%c0%af 就是这种字符串。这种类型的攻击可使攻击者遍历远程系统目录结构，获取宝贵的服务器或网络信息，甚至远程运行一个程序。

3. 攻击安全账户管理器文件

通过攻击安全账户管理器（SAM）文件，攻击者有可能获取对用户名和密码的访问权。一旦攻击者能够访问该信息，就能利用这些信息获取明显合法的网络资源访问权限。因此，管理 SAM 文件是防范攻击的一个重要步骤。管理方法包括：

- 利用系统密钥（syskey）在 SAM 文件上启用附加加密。

- 通过某项策略禁用局域网（LAN）Manager 身份验证及 LAN Manager 散列存储，并利用其他形式的身份验证（比如证书和生物检测）。
- 建立并执行复杂的密码策略。

4．缓冲区溢出

缓冲区溢出是攻击者为获取系统访问权限所采用的一种非常危险的技术。攻击者们会企图将过多信息放进一个容器，以观察它们能否获得以有意义的方式执行的溢出。比如，如果被攻击的程序没有执行适当的边界检查，那么它就会溢出，并允许攻击者执行他们选择的功能。这些溢出通常在具有完全管理权限的本地系统账户的环境内运行。

这些攻击最常见的类型就是基于堆栈的缓冲区溢出攻击。溢出会改写整个堆栈，包括指针。攻击者通过调整放在溢出中的数据量来利用这一弱点，然后发送执行某种命令的计算机特定代码和返回指针的一个新地址，最后在系统返回到堆栈时，利用该地址（回头指向堆栈）执行他们自己的程序指令。

5．拒绝服务攻击

攻击者不一定需要有系统访问权限才能产生巨大问题。拒绝服务（Denial of Service，DoS）攻击会耗尽系统资源，导致它不能执行正常的功能。例如，用尽某一服务器上的所有网络连接，或让邮件服务器必须处理超过其设计处理能力的大量邮件。DoS 攻击可能是某一直接攻击的结果，也可能是病毒、蠕虫或特洛伊木马导致的。

分布式拒绝服务（Distributed Denial of Service，DDoS）攻击会在攻击前在不同计算机上安装僵尸程序，以后向这些僵尸程序发出命令，再由僵尸程序代表攻击者发动攻击，因而隐藏了其踪迹。僵尸程序本身通常是利用蠕虫安装的。

DDoS 攻击的真正危险在于攻击者使用许多无辜的计算机作为主机来控制发动攻击的其他僵尸程序。当被攻击的系统试图追溯攻击时，它会收到由一系列僵尸程序产生的一组哄骗地址。

6．后门攻击

为防止攻击者下载系统信息，必须防范攻击者利用特洛伊木马在系统上安装后门。这通常在客户计算机上较为严重，而不是在得到全面保护的服务器上。但是，攻击者可以利用这种机制攻击用户或管理员的工作站，然后利用该系统对周边网络上发动攻击。可以采用以下方法来防止。

- 运行完整的病毒扫描，并用最新的特征文件及时更新防病毒工具。
- 当心通过电子邮件发送的所有内容，并限制未知附件的执行。
- 运行 Internet Security Scanner（ISS）扫描程序等工具，扫描整个网络是否存在攻击者工具，例如 Back Orifice，确保扫描程序数据库得到及时更新。
- 仅接受已签名的 Microsoft ActiveX 控件。

7．恶意代码

任何可执行代码都是一个潜在的风险。恶意代码的形式可以是在本单位内和单位与单

位之间扩散的破坏性代码（比如通过电子邮件），也可能是从本单位内部怀有恶意而故意运行的代码。

恶意代码可以概括地分为如下 4 种主要类型。

- 病毒。
- 蠕虫。
- 特洛伊木马。
- 其他恶意代码。

1.12 网络安全的关键技术

1. 防电磁辐射

防电磁辐射分为对传导发射的防护和对辐射的防护。

2. 访问控制技术

主要指网络、数据库、操作系统、应用程序和远程拨入等的访问控制。

3. 安全鉴别技术

安全鉴别技术包括如下几方面。

- 网络设备的鉴别，基于 VPN 设备和 IP 加密机的鉴别。
- 应用程序中的个人身份鉴别。
- 远程拨号访问中心加密设备（线路密码机）与远程加密设备（线路密码机）的鉴别。
- 密码设备对用户的鉴别。

4. 权限控制

权限控制包括操作系统、数据库的访问、密码设备管理和应用业务软件操作的权限控制。

5. 通信保密

通信保密包括如下内容。

- 中心网络中采用 IP 加密机进行加密。
- 远程拨号网络中采用数据密码机进行线路加密。

6. 数据完整性

使用消息完整性编码（Message Integrity Code，MIC）是一种 HASH 函数，用来计算信息的摘要。

7. 实现身份鉴别

可利用：

- 数字签名机制。

- 消息鉴别码。
- 校验等。
- 口令字。
- 智能卡。
- 身份验证服务，包括 Kerberos 和 X.509 目录身份验证。

8．安全审计

采用专用、通用设备相结合的方式，从以下几个方面着手：

- 数据库/操作系统的审计。
- 防火墙的进出数据审计。
- 应用业务软件/平台（如 Lotus 等）的审计安全备份。

9．病毒防范及系统安全备份

包括病毒的检测和防范，常规的系统安全备份。

10．加密方法

公共密钥加密和专用密钥加密，后又出现了椭圆曲线密码学。

11．网络的入侵检测和漏洞扫描

入侵检测的 5 种技术。

- 基于应用的监控技术。
- 基于主机技术的监控。
- 基于目标的监控技术。
- 基于网络的监控技术。
- 综合上述 4 种方法进行监控。

漏洞检测的 5 种技术。

- 基于应用的检测技术。
- 基于主机的检测技术。
- 基于目标的检测技术。
- 基于网络的检测技术。
- 综合上述 4 种方法进行检测。

注意：具体实现与网络拓扑结构有关。一般系统在网络系统中可设计为两个部分：安全服务器和侦测代理。分布在网上的代理大体上有三种：主机侦测代理、网络设备侦测代理和公用服务器侦测代理。

12．应用系统安全

总体思想是利用安全 PC 卡、PC 安全插卡来实现所有应用的安全保密。

13．文件传送安全

要求对等实体鉴别、数据加密、完整性检验和数字签名。

14．邮件安全

邮件加密、数字签名和利用安全 PC 卡。

1.13　保证网络安全的措施

保证网络安全的措施一般包括如下几种。

1．防火墙

防火墙是一种防御技术，它在网络安全中是不可缺少的。它像一道防盗门，把内部网和外部网分隔开来，转发可信的分组数据包，丢弃可疑的数据包。将在第 8 章中详细讲述防火墙。

2．身份认证

身份认证是一致性验证的一种，验证（identification）是建立一致性证明的一种手段。身份验证主要包括验证依据、验证系统和安全要求。

3．加密

加密是通过对信息的重新组合，使得只有收发双方才能解码还原信息。传统的加密系统是以密钥为基础的，这是一种对称加密，也就是说，用户使用同一个密钥加密和解码。

目前，随着技术的进步，加密已被集成到系统和网络中，如 Internet Engineering Task Fore，正在发展的下一代网际协议 IPv6。硬件方面，Intel 公司也在研制用于 PC 和服务器主板的加密协处理器。按作用不同，数据加密技术主要分为数据传输、数据存储、数据完整性的鉴别及密钥管理技术 4 种。

1）数据传输加密技术

目的是对传输中的数据流加密，常用的方针有线路加密和端到端加密两种。前者侧重于线路上而不考虑信源与信宿，对保密信息通过各线路采用不同的加密密钥提供安全保护。后者则指信息由发送端自动加密，并进入 TCP/IP 数据包回封，然后作为不可阅读和不可识别的数据经过因特网，这些信息一旦到达目的地，将被自动重组、解密，成为可读数据。

2）数据存储加密技术

目的是防止在存储环节上的数据失密，可分为密文存储和存取控制两种。前者一般是通过加密算法转换、附加密码和加密模块等方法实现；后者则是对用户资格加以审查和限制，防止非法用户存取数据或合法用户越权存取数据。

3）数据完整性鉴别技术

目的是对介入信息的传送、存取和处理的人的身份和相关数据内容进行验证，达到保密的要求。一般包括口令、密钥、身份和数据等项的鉴别，系统通过对比验证对象输入的

特征值是否符合预先设定的参数，实现对数据的安全保护。

4）密钥管理技术

为了数据使用的方便，数据加密在许多场合集中表现为密钥的应用，因此密钥往往是保密与窃密的主要对象。密钥的媒体有磁卡、磁带、磁盘和半导体存储器等。密钥的管理技术包括密钥的产生、分配保存、更换与销毁等各环节上的保密措施。

4．数字签名

大多数电子交易采用两个密钥加密：密文和用来解码的密钥一起发送，而该密钥本身又被加密，还需要另一个密钥来解码。这种组合加密被称为数字签名，它有可能成为未来电子商务中首选的安全技术。

美国政府的加密标准 DSS（Digital Signature Standard），使用了 Secure Hash 运算法则。用该法则对信息进行处理，可得到一个 160 位的数字，把这个数字以某种方式与信息的密钥组合起来，从而得到数字签名。完整性是在数据处理过程中，在原来数据和现行数据之间保持完全一致的证明手段。

现在比较普遍采用的签名算法有 RSA 和 DSS。

5．内容检查

即使有防火墙、身份认证和加密，人们仍然担心遭到病毒的攻击。这些病毒通过 E-mail 或用户下载的 Java 和 ActiveX 小程序（Applet）进行传播。带病毒的 Applet 激活后，又可能会自动下载别的 Applet。现有的反病毒软件可以清除 E-mail 病毒，而对付 ActiveX 病毒也有一些办法，如完善防火墙，使之能够监控 Applet 的运行，或者给 Applet 加上标签，让用户知道它们的来源。

6．存取控制

存取控制规定何种主体对何种客体具有何种操作权力。存取控制是内部网安全理论的重要方面，主要包括人员限制、数据标识、权限控制、控制类型和风险分析。

7．安全协议

安全协议的建立和完善是安全保密系统走上规范化、标准化道路的基本因素。根据计算机专用网多年的经验，一个较为完善的内部网和安全保密系统，至少要实现加密机制、验证机制和保护机制。目前，已开发并应用的有如下几种协议。

1）加密协议

加密协议有两个要素，一是能把保密数据转换成公开数据，在公用网中自由发送；二是能用于授权控制，无关人员无法解读。因此，数据要划分等级，算法也要划分等级，以适应多级控制的安全模式。

2）身份验证协议

身份验证是上网的第一道关口，且与后续操作相关，因此身份验证至少应包括验证协议和授权协议。人员要划分等级，不同等级具有不同的权限，以适应多级控制的安全模式。

3）密钥管理协议

包括密钥的生成、分发、存储、保护和公证等协议，保证在开放环境中灵活地构造各种封闭环境。根据因特网的特点，密钥分离度在网上要做到端、级和个人级，在库中要做到字节级。

4）数据验证协议

包括数据压缩、数据验证和数字签名。数字签名要同时具有端、级签名和个人签名的功能。

5）安全审计协议

包括与安全有关的事件，包括事件的探测、收集和控制，能进行事件责任的追查。

6）防护协议

除了采用防病毒卡、干扰仪等物理性防护措施外，还对用于信息系统自身保护的数据（审计表等）和各种秘密参数（用户口令、密钥等）进行保护，以增强反入侵功能。

8. 智能卡技术

与数据加密技术紧密相关的另一项技术则是智能卡技术。所谓智能卡就是密钥的一种媒体，一般就像信用卡一样，由授权用户所持有并由该用户赋予它一个口令或密码字，该密码与内部网络服务器上注册的密码一致。当口令与身份特征共同使用时，智能卡的保密性能还是相当有效的。

1.14 网络的安全策略

纵深防御策略可以保护资源免受来自外部和内部的威胁。因为网络安全是各个部分安全的总和，所谓纵深防御，就是以网络周边为起点，从外部路由器一直到资源子网中所在的位置中所有相关层面的保护。

通过多层安全保护，可以确保即使某一环节遭到破坏，而其他层仍能提供保护资源所需的安全。如图 1-4 所示为一个有效的纵深防御策略。



图 1-4

下面介绍图 1-4 中的各个组成部分。

1.14.1　数据防御

一些公司及科研院所最宝贵的资源之一是数据。如果这些数据落入竞争者的手中，或者数据被破坏，那么后果是不堪设想的。

对于客户端来说，存储在本地的数据更易受攻击。如果罪犯窃取了一台笔记本式计算机，即使他无法登录到该系统，也可将其中的数据备份，重新存储到其他地方并读取这些数据。

有多种方法可以保护数据，包括使用加密文件服务（EFS）或第三方加密工具对数据进行加密及修改文件中的自由访问控制列表。

1.14.2　应用程序防御

作为防御策略中的另一层，应用程序增强策略是所有安全模型的重要部分。应用程序存在于系统环境中，因此每一个应用程序，在获准应用于产品环境之前，都应在测试环境中对它们进行严格的安全达标测试。

1.14.3　主机防御

对环境中的每一台主机进行评估，然后制定相应的策略以限制每台服务器仅能执行规定的任务。这一做法相当于又设置了一道安全屏障，攻击者要想制造任何破坏就需要先逾越这道屏障。

方法之一是根据每台服务器所包含的数据的分类和类型制定各自的策略。例如，某一单位的策略可以规定所有的 Web 服务器都是面向公众的，因此，在这些服务器上只能包含公共信息。企业的数据库服务器被视为公司机密，这意味着要不惜一切代价保护这些服务器中的信息。

1.14.4　网络防御

网络防御主要是转发网络上的合法通信并阻止所有不需要的通信。可以使用 IPSec 对内部网络上的数据包进行加密，并将 SSL 用于外部通信。

1.14.5　周边防御

保护网络周边是抵御外界攻击的最重要内容。如果网络周边很安全，那么内部网络就会得到保护，不会遭受外部攻击。防火墙是网络周边防御的重要组成部分。安装一道或多道防火墙，以确保最大限度地减少外部攻击，并利用审计和入侵检测功能确保自己在攻击发生后及时发现情况。

1.14.6 物理安全

把物理安全作为整体安全策略的基石。保护服务器所在地点的物理安全是首要任务，保护范围包括服务器机房或整个数据中心。

作为风险管理策略的组成部分，应该确定适合于环境的物理安全级别。可以采取的物理安全措施包括以下部分或全部措施：

- 采取物理措施保护办公楼的所有区域（包括钥匙卡、生物检测设备和安全保卫）。
- 要求客人在到达时登记所有计算设备。
- 要求所有员工登记他们所拥有的任何便携式设备。
- 在将数据存储设备从办公楼搬走之前，都要对它们一一进行登记。
- 将服务器放在仅有管理员才允许进入的单独房间等。

1.15 网络攻击常用工具

当网络管理员懂得常用攻击工具的功能和使用时，就更能知己知彼，制定更佳的安全策略来应对相应的攻击工具。下面分别进行介绍。

任何成功的攻击都需要首先进行嗅探（sniffing）或端口扫描（port scanning）。首先介绍在第一步中使用的网络攻击工具。

网络嗅探的常用工具为 Ethereal，它是开放源代码的软件，免费的网络协议检测程序，支持 UNIX、Windows。感兴趣的读者可以从 www.Ethereal.com/网站或国内的 http://www.52z.com/Down/5664.html 中进行下载。Ethereal 是最流行的网络协议分析器，主要用于网络中数据包的监控，并且它依附于相应的网络。可以对网络中大部分协议的数据包进行追踪，包括每一个包流向和内容等，可以方便地查看及监控 TCP 会话动态。具体的使用过程如下。

（1）首先需要安装 winpcap，然后才能使用 Ethereal。winpcap 的下载地址为 http://netgroup.org，或者是国内的 www.skycn.com 等软件下载网站。winpcap 是用于 Windows 环境中网络链路层访问的标准工具，可以捕获通过协议栈的所有应用数据包，还可以对核心级数据包进行过滤，同时可以截获远程数据包。

（2）安装 Ethereal。

（3）启动 Ethereal 以后，选择菜单栏中的 Capture→Start 命令。图 1-5 是 Ethereal 中显示的截获数据包。

（4）一般还需要设置 Capture Options 对话框中的各选项，如图 1-6 所示。在 Interface 下拉列表中，选择需要指定的接口，即网卡。其余的可以选择默认。

（5）如果需要设置包过滤器，也就是截获用户所需要类型的数据包，可以在 Filter 中输入数据包的类型，例如截获 TCP 数据包，则在 Filter 文本框中输入 tcp，然后单击 Apply 按钮。截获的数据包如图 1-7 所示。文本框中输入值的内容用 C 语言来描述。读者可以查询其手册进行进一步学习。

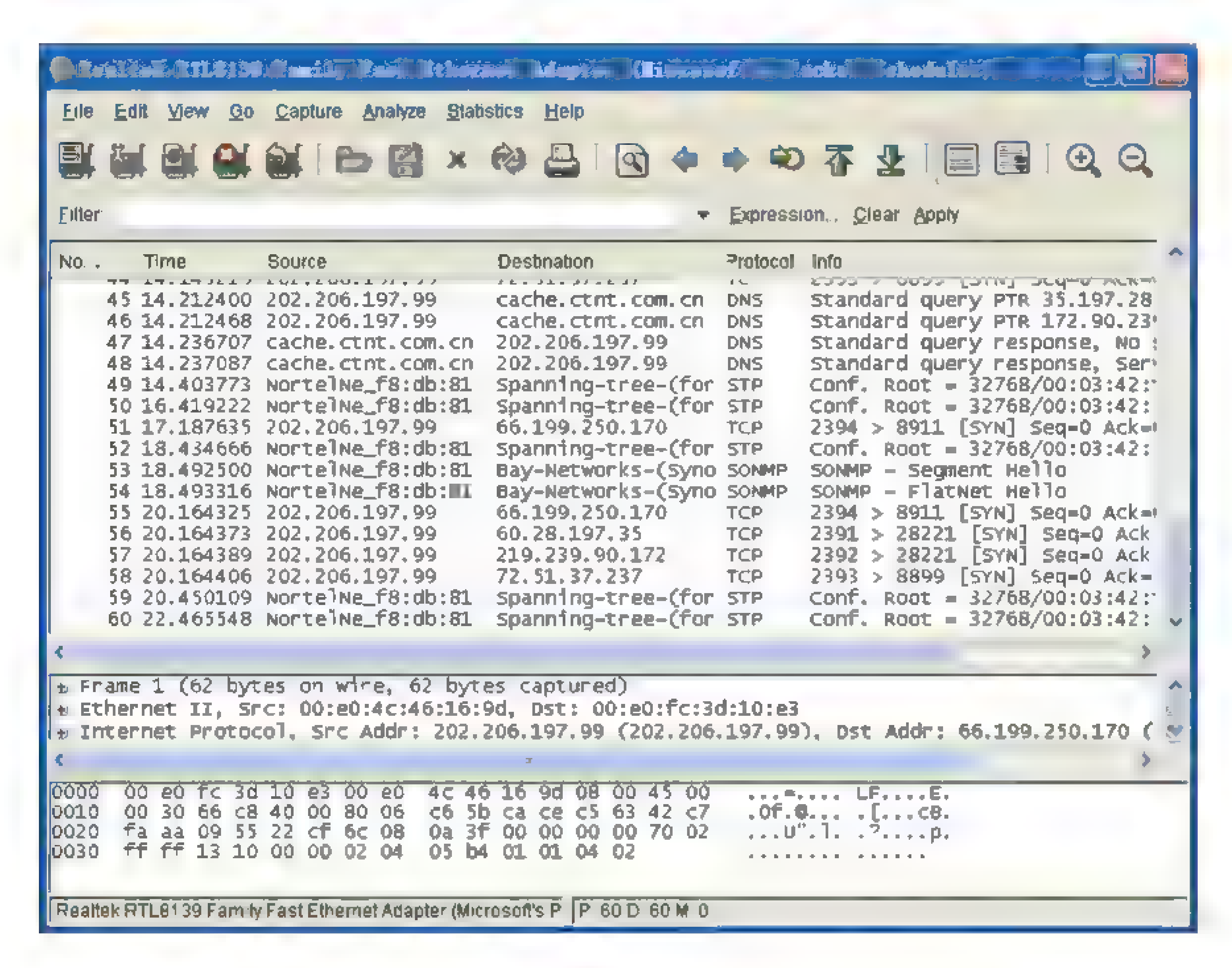


图　1-5

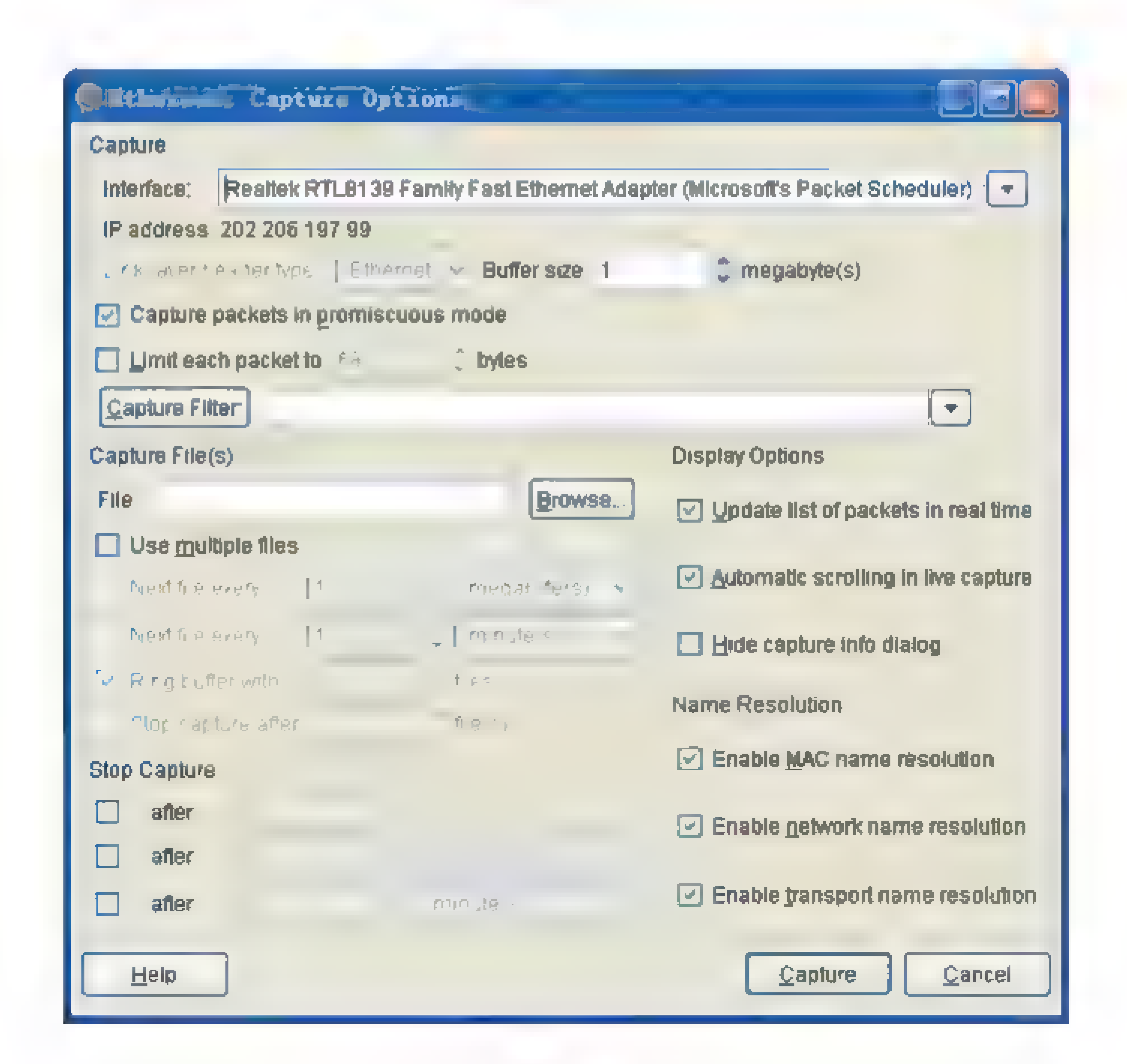


图　1-6

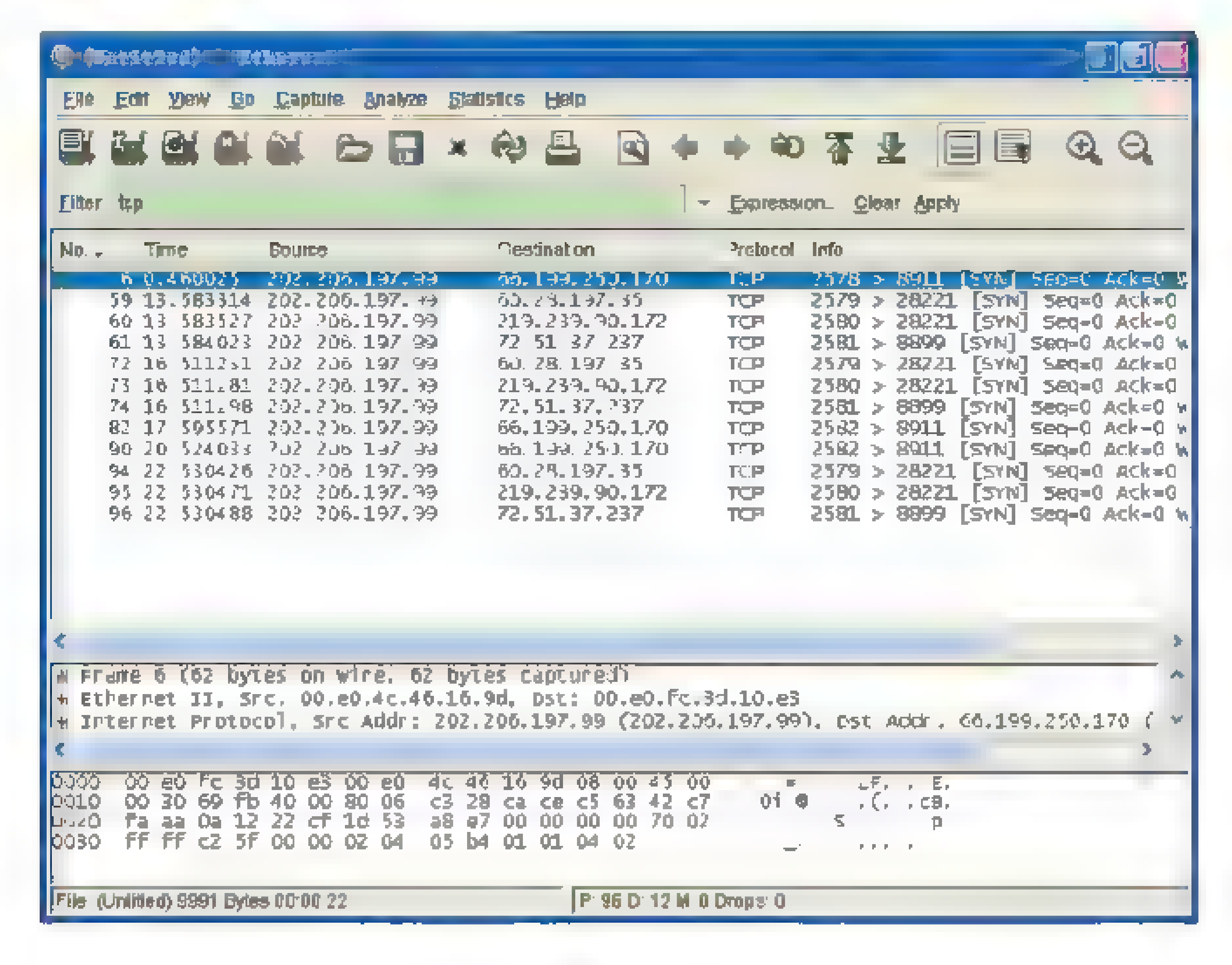


图 1-7

因为Ethereal是数据包嗅探器，它只是收集网络的信息，而并非主动进行攻击，本身是被动的，所以人们很难检测这类程序。

而端口扫描是主动的，常用的工具是nmap，感兴趣的读者可以从www.insecure.org/nmap网站下载。此工具是基于Linux和UNIX操作系统的，在第5章中详细介绍了nmap的使用。

另外，常用的应用程序Nessus也是用于扫描Apache的漏洞的。可以在www.nessus.org网站上下载。

第 2 章

加 密 技 术

在现代社会，随着因特网的发展，人们越来越多地在网络上从事个人事务处理和商业活动，网络已经发展成为人们日常生活和工作的重要工具。如使用网络发送电子邮件进行电子购物、资金转账、发布公告和接收重要数据等。信息安全性也变得越来越重要，主要表现有以下 4 个特性。

- 保密性（confidentiality）：保证信息不泄露给未经授权的任何人。
- 完整性（integrity）：防止信息被未经授权的人篡改。
- 可用性（availability）：保证信息和信息系统确实为授权者所用。
- 可控性（controllability）：对信息和信息系统实施安全监控，防止非法利用信息和信息系统。

加密技术作为一种主动的防卫手段，是网络安全最有效的技术之一。一个加密网络，不但可以防止非授权用户的搭线窃听和入网，而且也是对付恶意软件的有效方法。密码实现的是一种变换，利用密码变换保护信息秘密是密码最原始的能力，随着信息技术发展起来的现代密码学，不仅被用于解决信息的保密性，而且也用于解决信息的完整性、可用性和可控性。密码是解决信息安全的最有效手段，密码技术是解决信息安全的核心技术。在本章中将对加密技术的算法及密钥管理机制等进行详细的介绍。

2.1 密 码 算 法

密码算法是一个函数变换，要加密的信息称为明文，经过以密钥为参数的函数加以转换。加密过程输出的即是密文。加密公式可表示为 $C=EK(P)$，其中 C 代表密文，即加密后得到的字符序列；P 代表明文，即需要加密的字符序列；E 代表加密算法；K 表示密钥，是秘密选定的一个字符序列。

加密和解码的技术统称为密码学。密码学的原则是“一切秘密寓于密钥之中”，算法可以公开。当加密完成后，可以将密文通过不安全渠道送给收信人，只有拥有解密密钥的收信人可以对密文进行解密，即破译得到明文，密钥的传递必须通过安全渠道。目前流行的密码算法主要有 DES、RSA、IDEA 和 DSA 等。

密码算法可分为传统密码算法和现代密码算法。传统密码算法的特点是加密和解密必

须是同一密钥，如DES和IDEA；现代密码算法将加密密钥与解密密钥区分开来，且只有加密密钥事实上求不出解密密钥。因此传统密码算法又称对称密码算法（Symmetric Crypto-graphic Algorithms），现代密码算法称非对称密码算法或公钥密码算法（Public-Key Crypto-graphic Algorithms），是由Diffie和Hellman在1976年的美国国家计算机会议上提出这一概念的。按照加密时对明文的处理方式，密码算法又可分为分组密码算法和序列密码算法。分组密码算法是把密文分成等长的组分别加密，序列密码算法是一位一位地处理，用已知的密钥随机序列与明文按位异或。

在计算机系统中使用对称密钥密码体制已有多年，既有比较简便可靠的、久经考验的方法，如以DES（数据加密标准）为代表的分块加密算法；也有一些新的方法，如由RSA公司的Rivest研制的专有算法RC2、RC4和RC5等。其中RC2和RC5是分块加密算法，RC4是数据流加密算法。

密码算法主要有以下三类。

- 对称加密算法：这种技术使用单一的密钥加密信息。
- 公钥加密算法（非对称加密）：这种技术使用两个密钥，一个公钥用于加密信息，另一个私钥用于解密信息。
- 单项函数算法：这个函数对信息进行加密产生原始信息的一个“签名”，该签名被在以后证明它的权威性。这一般也称为数字签名。

下面介绍相关的算法。

2.2　对称加密技术

对称加密算法是一种传统的加密算法，它的基本原理如下：在对称加密中，数据信息的传送、加密及接收解密都需用到一个共享的钥匙，也就是说加密和解密共用一把钥匙。

例如，Mary想送一张订单给Jack，Mary希望只有Jack可以订它。Mary将这张订单（里面的文字）用一个加密钥匙加密之后，将这个加过密的订单（密码文字）寄给了Jack，然后Jack用同一把密钥对订单解密。

对称加密最常用的一种方式是资料加密标准（DES）。所有的参与者都必须彼此了解，而且完全互相信任，因为他们每一个人都有一份钥匙的珍藏副本。如果传送者和接收者位于不同的地点，无论他们在开面对面的会议，还是在公共传输系统（电话系统或邮局服务）上，当秘密钥匙被互相交换时，只要有人在钥匙传送的途中窃听或者拦截，这个人（黑客）就可以用这个钥匙来读取所有正在传输的已加密的数据信息，所以对称加密有很大的不安全性，容易被黑客所利用。

2.2.1　DES算法

在对称算法中，DES（Data Encryption Standard）算法是最著名的对称密钥加密算法。DES使用56位密钥对64位的数据块进行加密，并对64位的数据块进行16轮编码。在每轮编码时，一个48位的“每轮”密钥值由56位的完整密钥得出来。

DES的高速简便性使之流行，但是在现代网络中，信息安全越来越重要，由于DES

只有一个共享密钥，在网络传输过程中极容易被截获，造成在网络信息传输中的不安全性。所以现在非对称加密技术很流行，而现代网络中往往将对称和非对称加密技术相结合，达到取长补短。

2.2.2　三重 DES 算法

DES 的缺点就是密钥长度相对比较短，因为人们并没有放弃使用 DES，所以想出了一个解决其长度问题的方法，即采用三重 DES。这种方法用两个密钥对明文进行三次加密，假设两个密钥是密钥 1 和密钥 2，其算法的步骤如图 2-1 所示。

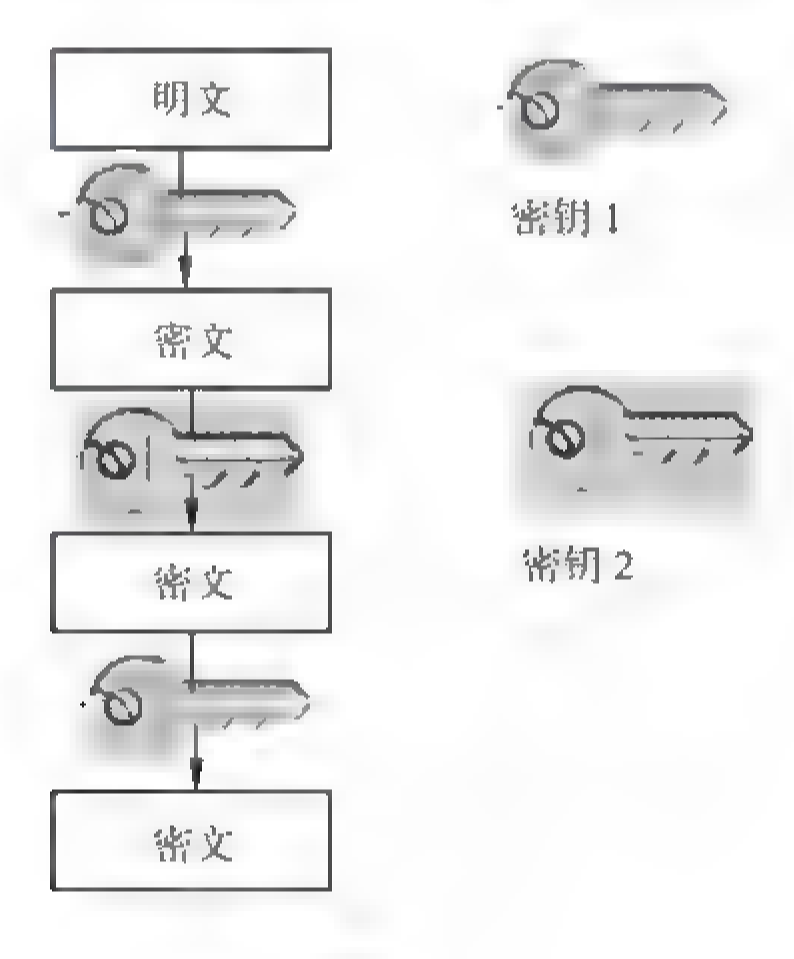


图　2-1

- 用密钥 1 进行 DES 加密。
- 用密钥 2 对步骤 1 的结果进行 DES 解密。
- 对步骤 2 的结果使用密钥 1 进行 DES 加密。

2.3　不对称加密技术

在现代网络的各种加密技术中，不对称加密技术（公钥）是其中一颗闪亮的明珠，也是最流行的加密技术，下面来讲述不对称加密技术。

在非对称加密算法中，利用了两把钥匙：一把钥匙用来将数据信息加密，而用另一把不同的钥匙来解密。这两把钥匙之间具有数学关系，所以用一个钥匙加密过的资料只能用相应的另一个钥匙来解密。非对称加密异于两方都用同一个密钥的对称加密算法，公钥密码法对每一个人都使用一对钥匙，其中一个是公开的，而另一个是私密的。公钥（公共密钥）可以让其他人知道，而私钥（专用密钥）则必须加以保密，只有持有人知道它的存在，但这两种钥匙都必须加以保证防止被修改。也就是每个人都有一对密钥，一个私钥和一个公钥，它们在数字上相关，在功能上不同。一个密钥锁上的用另一个可以打开，此技术使用两个加密的密钥来保证会话的安全。公钥可以给任何请求它的应用程序或用户，私钥只

有它的所有者知道。公钥加密算法也称非对称密钥算法，用两对密钥：一个公共密钥和一个专用密钥。用户要保障专用密钥的安全，公共密钥则可以发布出去。公共密钥与专用密钥是有紧密关系的，用公共密钥加密的信息只能用专用密钥解密，反之亦然。由于公钥算法不需要联机密钥服务器，密钥分配协议简单，所以极大地简化了密钥管理。除加密功能外，公钥系统还可以提供数字签名。

公钥加密算法中使用最广的是RSA。RSA使用两个密钥：一个公共密钥，一个专用密钥。如用其中一个加密，则可用另一个解密，密钥长度从40～2048位可变。加密时也把明文分成块，块的大小可变，但不能超过密钥的长度，RSA算法把每一块明文转化为与密钥长度相同的密文块。密钥越长，加密效果越好，但加密解密的开销也大，所以要在安全与性能之间折中考虑，一般64位较合适。RSA有一个比较知名的应用是SSL，在美国和加拿大SSL用128位RSA算法，由于出口限制，在其他地区（包括中国）通用的则是40位版本。

公共密钥加密算法主要有如下两种用途。

- 数据加密：发送者用接收者的公钥对要发送的数据加密，接收者用自己的私钥对接收到的数据解密。第三者由于不知道接收者的私钥而无法破译该数据。
- 身份认证：发送者可以用自己的私钥对要发送的数据加上“数字签名”，接收者通过验证“数字签名”就可以准确地确定数据的来源。

公共密钥加密算法又称为非对称加密算法，常见的加密算法有RSA、DSA等。

公共密钥方案较专用密钥方案处理速度慢，因此，通常把公共密钥与专用密钥技术结合起来实现最佳性能。即用公共密钥技术在通信双方之间传送专用密钥，而用专用密钥对实际传输的数据加密解密。

在Internet中使用最多的是公钥系统。即公开密钥加密，它的加密密钥和解密密钥是不同的。一般对于每个用户生成一对密钥后，将其中一个作为公钥公开，另外一个则作为私钥由属主保存。常用的公钥加密算法是RSA算法，加密强度很高。具体做法是将数字签名和数据加密结合起来。发送方在发送数据时必须加上数据签名，做法是用自己的私钥加密一段与发送数据相关的数据作为数字签名，然后与发送数据一起用接收方密钥加密。当这些密文被接收方收到后，接收方用自己的私钥将密文解密，得到发送的数据和发送方的数字签名，然后用发布方公布的公钥对数字签名进行解密，如果成功，则确定是由发送方发出的。数字签名每次还与被传送的数据和时间等因素有关。由于加密强度高，而且并不要求通信双方事先要建立某种信任关系或共享某种秘密，因此十分适合在Internet上使用。

公共密钥的优点就在于，也许你并不认识某一实体，但只要你的服务器认为该实体的CA（认证机构）是可靠的，就可以进行安全通信，而这正是电子商务这样的业务所要求的。例如信用卡购物。服务方对自己的资源可根据客户CA的发行机构的可靠程度来授权。目前国内外尚没有可以被广泛信赖的CA。美国Natscape公司的产品支持公共密钥，但把Natscape公司作为CA。可靠的CA中心有Verisign，可以登录它的网站进行证书的申请，网站地址为http://w12.263.net/www.verisign.com。

2.4　RSA 算法简介

RSA 算法既能用于数据加密也能用于数字签名，它易于理解和操作，也很流行。算法的名字以 R.Rivest、A.Sham ir 和 L.Adleman 三位发明者的名字命名。RSA 算法是第一个能同时用于加密和数字签名的算法，也是被研究得最广泛的公钥算法，从提出到现在已近 20 年，经历了各种攻击的考验，逐渐被人们接受，普遍认为是目前最优秀的公钥方案之一。

RSA 算法研制的最初理念与目标是努力使因特网安全可靠，旨在解决 DES 算法秘密密钥利用公开信道传输分发的难题。而实际结果是不但很好地解决了这个难题，还可利用 RSA 来完成对电文的数字签名以对抗电文的否认与抵赖。同时还可以利用数字签名较容易地发现攻击者对电文的非法篡改，以保护数据信息的完整性。所有这些在下文中均要详细介绍，首先介绍 RSA 算法。

2.4.1　RSA 算法

1. RSA 算法

（1）选两个大素数 r_1 和 r_2，通常均大于 10^{100}。

（2）计算 $n=r_1 \cdot r_2$ 和 $x=(r_1-1)(r_2-1)$（注：x 是欧拉函数）。

（3）选一个与 x 互质的数，令其为 d。

（4）找到一个 e，满足 $e \cdot d=1(\mathrm{mod}\ x)$。

（5）选好这些参数后，因为 RSA 是一种分组密码系统，所以先将明文划分成块，使得每个明文报文 P 的长度 m 满足 $0<m<n$。加密 P 时计算 $C=P^d\ (\mathrm{mod}\ n)$，解密 C 时计算 $P=C^d\ (\mathrm{mod}\ n)$。由于模运算的对称性，可以证明加密解密在一定范围内是可逆的。

RSA 加密算法使用了两个非常大的素数来产生公钥和私钥。即使从一个公钥中通过因数分解可以得到私钥，但这个运算所包含的计算量是非常巨大的，以至于在现实上是不可行的。加密算法本身也是很慢的，这使得使用 RSA 算法加密大量的数据变得有些不可行。所以在大量数据进行加密传输时一般采用非对称算法（RSA 等）和对称算法结合的方法。如 PGP 算法（及大多数基于 RSA 算法的加密方法）使用公钥来加密一个对称加密算法的密钥，然后再利用一个快速的对称加密算法来加密数据。这个对称算法的密钥是随机产生的，是保密的，因此得到这个密钥的唯一方法就是使用私钥来解密。

2. RSA 的具体工作原理

主机 A 和主机 B 进行安全的数据传输，那么首先主机 A 随机产生密钥 1，并使用主机 B 的公钥进行加密，然后发送给主机 B，主机 B 使用自己的私钥进行解密，得到主机 A 的密钥；然后主机 B 随机生成密钥 2，使用主机 A 的公钥对密钥 2 进行加密后传送给主机 A。此时主机 A 和主机 B 都同时得到了密钥 1 和密钥 2，也就是彼此之间的私钥。因为在不对称加密算法中，公钥是公开的，在证书中就可以得到，其原理示意图如图 2-2 所示。

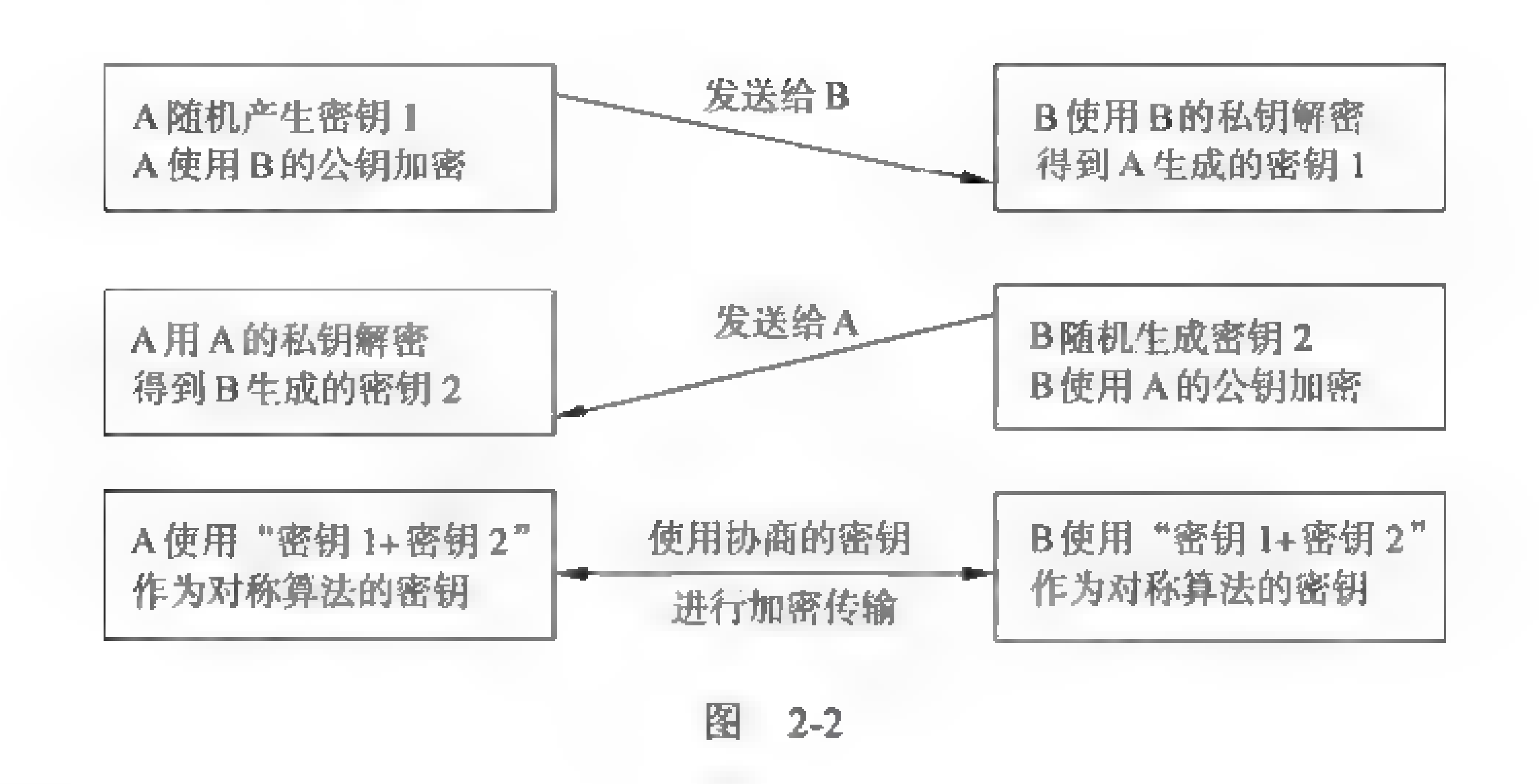


图 2-2

2.4.2 密钥对的产生

选择两个大素数 p 和 q，计算：

$n=p\cdot q$

然后随机选择加密密钥 e，要求 e 和（$p-1$）•（$q-1$）互质。最后，利用欧拉算法计算解密密钥 d，满足 $e\cdot d=1(\mathrm{mod}\ (p-1)\cdot(q-1))$，其中 n 和 d 也要互质。数 e 和 n 是公钥，d 是私钥。两个素数 p 和 q 不再需要，应该丢弃，不要让任何人知道。

加密信息 m（二进制表示）时，首先把 m 分成等长数据块 m_1，m_2，…，m_i，块长为 s，其中 $2^s<=n$，s 尽可能的大。对应的密文如下。

$c_i=m_i^e$（mod n） 公式（a）

解密时作如下计算。

$m_i=c_i^d$（mod n） 公式（b）

RSA 可用于数字签名，方案是用公式（a）签名，公式（b）验证。具体操作时考虑到安全性和 m 信息量较大等因素，一般是先做 HASH 运算。因为 HASH 是一类特殊的散列函数，它可以在传送的报文中提取发送者的特征数据，也就是生成摘要，这份摘要是独一无二的。因为私钥是唯一的，只有拥有者才知道，所以这一特征数据也是唯一的。这一特征数据被称为数字指纹，所以经 HASH 运算得到的摘要即数字指纹可以用于数字签名。

2.4.3 RSA 的安全性

RSA 的安全性依赖于大数分解，但是否等同于大数分解一直未能得到理论上的证明，因为没有证明，破解 RSA 就一定需要做大数分解。假设存在一种无须分解大数的算法，那它肯定可以修改成为大数分解算法。目前，RSA 的一些变种算法已被证明等价于大数分解。无论怎样，分解 n 是最明显的攻击方法。现在，人们已能分解 140 多个十进制位的大素数。因此，模数 n 必须选大一些，应视具体实用情况而定。

2.4.4 RSA 的速度

由于进行的都是大数计算，使得 RSA 最快的情况也比 DES 慢 100 倍，无论是软件还

是硬件实现，速度一直是 RSA 的缺陷，它一般只用于少量数据加密。

2.4.5　RSA 的选择密文攻击

RSA 在选择密文攻击面前很脆弱。一般攻击者是将某一信息做一下伪装（blind），让拥有私钥的实体签署，然后经过计算就可得到它所想要的信息。实际上，攻击利用的都是同一个弱点，即存在这样一个事实，乘幂保留了输入的乘法结构。

$(XM)^d=X^d*M^d \bmod n$

前面已经提到，这个固有的问题来自于公钥密码系统最有用的特征——每个人都能使用公钥。但从算法上无法解决这一问题，主要措施有两条：一条是采用好的公钥协议，保证工作过程中实体不对其他实体产生的任何信息解密，不对自己一无所知的信息签名；另一条是决不对陌生人送来的随机文档签名，签名时首先使用 One-Way HASH 函数对文档做 HASH 处理，或同时使用不同的签名算法。下面是几种不同类型的攻击方法。

1．RSA 的公共模数攻击

若系统中共有一个模数，只是不同的人拥有不同的 e 和 d，那么这个系统将是危险的。最普遍的情况是同一信息用不同的公钥加密，这些公钥共模而且互质，那么该信息无须私钥就可得到恢复。

设 P 为信息明文，两个加密密钥为 e_1 和 e_2，公共模数是 n，则：

$C_1 = P^{e_1} \bmod n$

$C_2 = P^{e_2} \bmod n$

密码分析者知道 n、e_1、e_2、C_1 和 C_2，就能得到 P。

因为 e_1 和 e_2 互质，故用欧拉算法能找到 r 和 s，满足：

$r*e_1+s*e_2=1$

假设 r 为负数，需再用欧拉算法计算 $C_1^{(-1)}$，则：

$C_1^{(-1)^{-r}} \square C_2^s = P \bmod n$

另外，还有其他几种利用公共模数攻击的方法。总之，如果知道给定模数的一对 e 和 d，一是有利于攻击者分解模数，二是有利于攻击者计算出其他成对的 e' 和 d'，而无须分解模数。解决办法只有一个，那就是不要共享模数 n。

2．RSA 的小指数攻击

有一种提高 RSA 速度的建议是使公钥 e 取较小的值，这样会使加密变得易于实现，速度有所提高。但这样做是不安全的，对付办法就是 e 和 d 都取较大的值。

2.4.6　RSA 的数字签名

公钥体系中有公钥和私钥，私钥保持私有，只有拥有者才知道；公钥广泛分布（通常作为公共证书的一部分），因此任何人都能用公钥加密数据，而只有私钥拥有者才能解密。另外，私钥拥有者用私钥加密数据，任何拥有公钥的人都能解开，这通常用作数字签名。

在这种情况下，签名者产生一个数字信息（例如 HASH）使用协商好的算法，然后用私钥加密。接收者能验证私钥拥有者发送的消息，用签名者的公钥解开加密的信息，并产生与收到信息相匹配的摘要。

RSA 公钥体系可以用于对数据信息进行数字签名。所谓数字签名就是信息发送者用其私钥对从所传报文中提取的特征数据或称数字指纹进行 RSA 算法解密运算操作，得到发信者对该数字指纹的签名函数 H(m)。签名函数 H(m)从技术上标识了发信者对该电文的数字指纹的责任。因发信者的私钥只有他本人才有，所以他一旦完成了签名便保证了发信人无法抵赖曾发过该信息（即不可抵赖性）。经验证无误的签名电文同时也确保了信息报文在经签名后未被篡改（即完整性）。当信息接收者收到报文后，就可以用发送者的公钥对数字签名的真实性进行验证。美国参议院已通过了立法，数字签名与手书签名的文件具有同等的法律效力。

在数字签名中有重要作用的数字指纹是通过一类特殊的散列函数（HASH 函数）生成的，对这些 HASH 函数的特殊要求是：

- 接收的输入报文数据没有长度限制。
- 对任何输入报文数据生成固定长度的摘要（数字指纹）输出。
- 从报文能方便地算出摘要。
- 难以对指定的摘要生成一个报文，而由该报文可以算出该指定的摘要。
- 难以对两个不同的报文生成相同的摘要。

2.4.7 RSA 的缺点

- 产生密钥很麻烦，受到素数产生技术的限制，因而难以做到一次一密。
- 分组长度太大，为保证安全性，n 至少也要 600 位以上，使运算代价很高。尤其是速度较慢，较对称密码算法慢几个数量级，并且随着大数分解技术的发展，这个长度还在增加，不利于数据格式的标准化。

2.4.8 关于 RSA 算法的保密强度安全评估

RSA 算法的保密强度，随其密钥的长度增加而增强。但是，密钥越长，其加解密所耗的时间也越长。因此，要根据所保护信息的敏感程度及攻击者破解所要花费的代价值不值得和系统所要求的反应时间来综合考虑决定，尤其是商业信息领域更是如此。

以下列出美国麻省理工学院的 RSA129（N=10^{129} 素因子分解攻击研究小组 Hal Abelson、Jeff schiller、Brian lamacchia 和 Derek Atkins）。根据他们对 PGP RSA（MPQS）算法攻击研究的结果如下：

```
RSA-129(429 -bit key)4600 MIPS-YEARS
```

这个表达式的含义是要 4600 台 VAX11/780 联合运行一年的时间，或一台 Pentium 运行 46 年时间才能将一个 N×10^{129} 的大数分解，找到其 P 和 Q。

下面列出 RSA 算法中密钥位数分别是 512、700、1024 和 384 的算法破解所需要的时

间，从中可以看出密钥的位数越长，其攻击者需要破解密钥的时间越长。

```
RSA 512 -bit key 42*10^4 MIPS-YEARS
RSA 700 -bit key 42*10^8 MIPS-YEARS
RSA 1024 -bit key 2.8*10^15 MIPS-YEARS
RSA 384 -bit key 470 MIPS-YEARS
```

公开密钥算法是在 1976 年由当时在美国斯坦福大学的迪菲（Diffie）和赫尔曼（Hellman）两人首先发明的（论文《New Direction in Cryptography》）。但目前最流行的 RSA 分别取自三名发明此算法的数学家（R.Rivest、A.Shamir 和 L.Adleman）的名字的第一个字母。

RSA 算法研制的最初理念与目标旨在解决 DES 算法秘密密钥利用公开信道传输分发的难题。而实际结果不但很好地解决了这个难题，还可利用 RSA 完成对电文的数字签名，以对抗电文的否认与抵赖。同时还可以利用数字签名较容易地发现攻击者对电文的非法篡改，以保护数据信息的完整性。

攻破 RSA 密钥的时间

密钥长度：100　200　300　500　750　1000

破解时间：30 秒　3 天　9 年　1 兆年　2×10^9 年　6×10^{15} 年

可以看出，当密钥长度大于 512 位时，以有限的人生攻破密钥是无法实现的。

2.4.9 RSA 的实用性

公开密钥密码体制与对称密钥密码体制相比较，确实有其不可取代的优点，但它的运算量远大于后者，超过几百倍、几千倍甚至上万倍，且复杂得多。

在网络上全都用公开密钥密码体制来传送机密信息，是没有必要的，也是不现实的。在计算机系统中使用对称密钥密码体制已有多年，既有比较简便可靠的、久经考验的方法，如以 DES（数据加密标准）为代表的分块加密算法（及其扩充 DESX 和 Triple DES）；也有一些新的方法发表，如 RSA 公司的专有算法 RC2、RC4 和 RC5 等。其中 RC2 和 RC5 是分块加密算法，RC4 是数据流加密算法。

在传送机密信息的网络用户双方，如果使用某个对称密钥密码体制（例如 DES），同时使用 RSA 不对称密钥密码体制来传送 DES 的密钥，就可以综合发挥两种密码体制的优点，即 DES 高速简便性和 RSA 密钥管理的方便和安全性。

RSA 算法已经在因特网中的许多方面得以广泛应用，包括在安全接口层标准（该标准是网络浏览器建立安全的因特网连接时必须用到的）方面的应用。

基于 RSA 算法的公钥加密系统具有数据加密、数字签名、信息源识别及密钥交换等功能，目前，RSA 加密系统主要应用于智能 IC 卡和网络安全产品。选用 RSA 算法作为公共钥加密系统主要算法的原因是算法安全性好。在模 N 足够长时，ln N 个整数中就有一个大小接近于 N 的素数。在模长为 1024 位时，可以认为 RSA 密码系统的可选密钥个数足够多，可以得到随机、安全的密钥对。公共钥加密系统多用于分布式计算环境，密钥分配和管理易于实现，局部攻击难以对整个系统的安全造成威胁。

2.5 RSA算法和DES算法的比较

DES数据加密标准用于对64位的数据进行加密和解密。DES算法所用的密钥也是64位，但由于其中包含了8个奇偶校验位，因而实际的密钥长度是56位。DES算法多次组合迭代算法和换位算法，利用分散和错乱的相互作用，把明文编制成密码强度很高的密文。DES算法的加密和解密的流程是完全相同的，区别仅仅是加密与解密使用子密钥序列的顺序正好相反。

RSA算法是公开密钥系统中的杰出代表，其算法的安全性是建立在具有大素数因子的合数的因子分解困难这一法则之上的。RSA算法中加密密钥和解密密钥不相同，其中加密密钥公开，解密密钥保密，并且不能从加密密钥或密文中推出解密密钥。

DES算法和RSA算法各有优缺点，可以在以下几个方面进行比较。

1. 在加密、解密的处理效率方面

DES算法优于RSA算法。因为DES密钥的长度只有56位，可以利用软件和硬件实现高速处理；RSA算法需要进行诸如200位整数的乘幂和求模等多倍字长的处理，处理速度明显慢于DES算法。

2. 在密钥的管理方面

RSA算法比DES算法更加优越。因为RSA算法可采用公开形式分配加密密钥，对加密密钥的更新也很容易，并且对不同的通信对象，只需对自己的解密密钥保密即可；DES算法要求通信前对密钥进行秘密分配，密钥的更换困难，对不同的通信对象，DES必须产生和保管不同的密钥。

3. 在安全性方面

DES算法和RSA算法的安全性都较好，目前还没有在短时间内破译它们的有效方法。

4. 在签名和认证方面

DES算法从原理上不可能实现数字签名和身份认证，但RSA算法能够容易地进行数字签名和身份认证。

总的来说，两种算法各具特点，DES算法加密、解密速度快，所以对数据量大、需要在网上传播的信息，用DES算法来加密和解密。

对于数据量小但非常重要的数据，数字签名和DES算法的密钥就要使用RSA算法进行加密和解密。

2.6 DSS/DSA算法

DSA（Digital Signature Algorithm）是Schnorr和ElGamal签名算法的变种，被美国NIST称为DSS（Digital Signatures Standard）。算法中应用了下述参数。

p：L位长的素数，L是64的倍数，范围是512～1024。

q：p –1 的 160 位的素因子。

g：g=h^((p –1)/q) mod p，h 满足 h<p–1，h^((p–1)/q) mod p>1。

x：x<q，x 为私钥。

y：y=g^x mod p，(p, q, g, y)为公钥。

H(x)：One-Way HASH 函数。DSS 中选用 SHA（Secure Hash Algorithm）。

p、q、g 可由一组用户共享，但在实际应用中，使用公共模数可能会带来一定的威胁。签名及验证协议如下：

（1）P 产生随机数 k，k<q。

（2）P 计算 r= (g^k mod p) mod q

s= (k^(–1) (H (m)+xr)) mod q

签名结果是(m, r, s)。

（3）验证时计算 w=s^ (–1) mod q

u1=(H (m)*w) mod q

u2= (r*w) mod q

v=((g^u1*y^u2) mod p) mod q

若 v=r，则认为签名有效。

DSA 是基于整数有限域离散对数问题的，其安全性与 RSA 相比差不多。DSA 的一个重要特点是两个素数公开，这样，当使用别人的 p 和 q 时，即使不知道私钥，也能确认它们是否是随机产生的，而这正好是优于 RSA 算法之处。

2.7　椭圆曲线密码算法

1985 年 N.Koblitz 和 V.Miller 提出将椭圆曲线用于密码算法，其根据是有限域上的椭圆曲线上点群中的离散对数问题 ECDLP。ECDLP 是比因子分解问题更难的问题，许多密码专家认为它有指数级的难度。从目前已知的最好求解算法来看，160 位的椭圆曲线密码算法的安全性相当于 1024 位的 RSA 算法，因此椭圆曲线上的密码算法速度很快。

图 2-3 为 RSA 算法和椭圆曲线密码算法的难度比较。

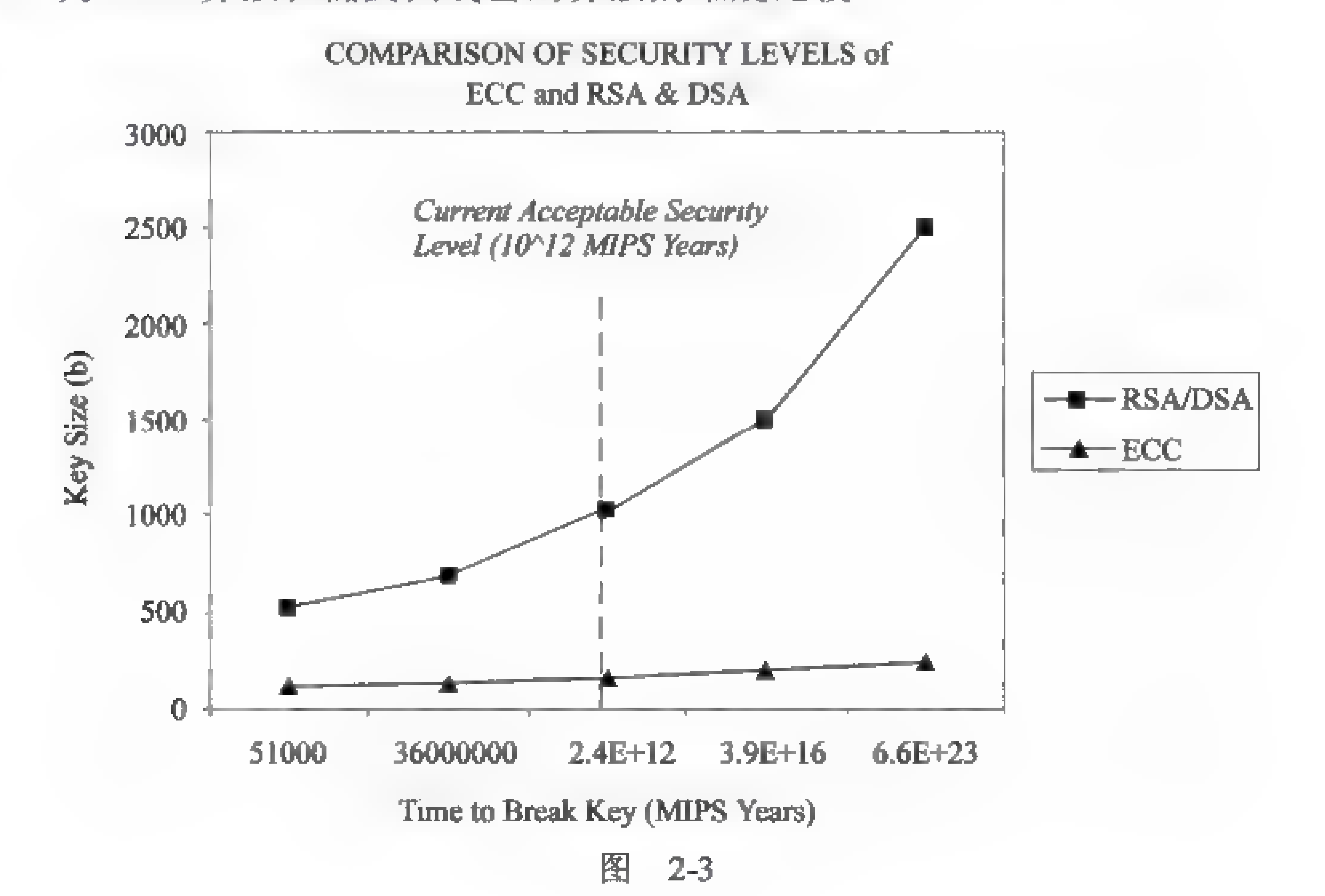


图　2-3

下面介绍椭圆曲线密码算法。

1．有限域上的椭圆曲线

设 K 表示一个有限域，E 是域 K 上的椭圆曲线，则 E 是一个点的集合。

$E/K=\{(x, y)|\ y^2+a_1xy+a_3y=x^3+a_2x^2+a_4x+a_6$

$a_1, a_3, a_2, a_4, a_6, x, y\in K\}\cup\{O\}$

其中 O 表示无穷远点。

在 E 上定义“+”运算，$P+Q=R$，R 是过 P、Q 的直线与曲线的另一交点关于 x 轴的对称点。当 $P=Q$ 时，R 是 P 点的切线与曲线的另一交点关于 x 轴的对称点。这样，$(E, +)$ 构成可换群（Abel 群），O 是加法单位元（零元）。

椭圆曲线离散对数问题 ECDLP 定义如下：给定定义在 K 上的椭圆曲线 E，一个 n 阶的点 $P\in E/K$ 和点 $Q\in E/K$，如果存在 l，确定整数 l，$0\leqslant l\leqslant n-1$，$Q=lP$。前面已经提到，ECDLP 是比因子分解难得多的问题。

图 2-4 显示了椭圆曲线上的加法：$P+Q=R$。

图 2-5 显示了椭圆曲线上一点的 2 倍：$P+P=R$。

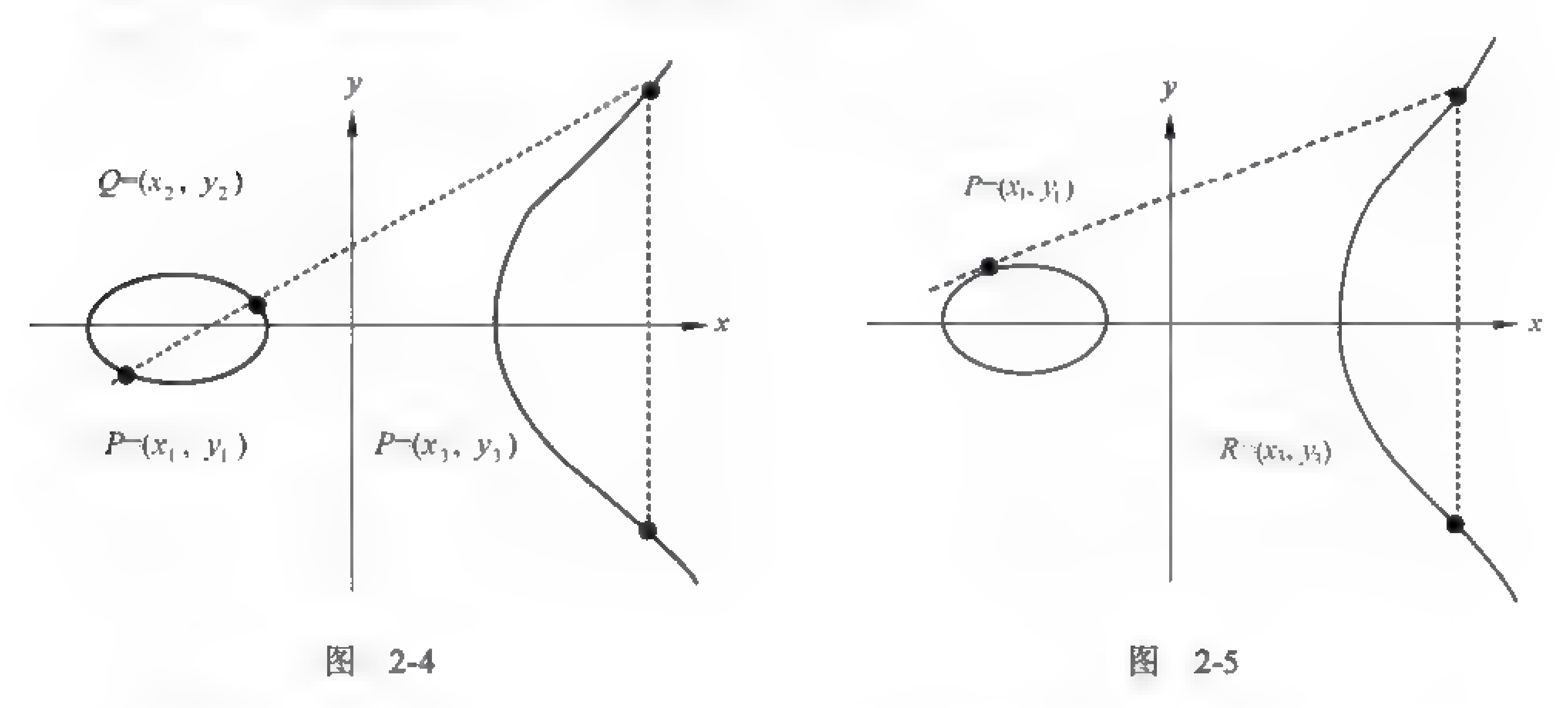


图 2-4　　　　图 2-5

2．椭圆曲线上的密码算法

基于该难题，N.Koblitz 和 V.Miller 在 1985 年分别利用有限域上椭圆曲线的点构成的群实现了离散对数密码算法，其中被广泛接受的是椭圆曲线上的 DSA，称为 ECDSA。随即展开了椭圆曲线密码学研究，除椭圆曲线外，还有人提出在其他类型的曲线，如超椭圆曲线上实现公钥密码算法。

此后，有人在椭圆曲线上实现了类似 ElGamal 的加密算法，以及可恢复明文的数字签名方案。除有限域上的椭圆曲线密码算法外，人们还探索了在椭圆曲线上实现 RSA 算法，如 KMOV 等。

3．椭圆曲线密码算法的发展

由于其自身优点，椭圆曲线密码学一出现便受到关注。现在密码学界普遍认为它将替

代 RSA 成为通用的公钥密码算法，SET（Secure Electronic Transactions）协议的制订者已把它作为下一代 SET 协议中默认的公钥密码算法，目前已成为研究的热点，是很有前途的研究方向。

2.8　量子加密技术

新出现的量子加密系统的基本原理是在 20 世纪 70 年代早期提出的，时间和公钥加密技术同步，直到现在，它的真正价值才得以体现。

下面介绍量子加密算法的工作原理：量子加密法是两个使用此加密算法的用户各自产生一个私有的随机数字符串。第一个用户向第二个用户的接收装置发送代表数字字符串的单个量子序列（光脉冲），接收装置从两个字符串中取出相匹配的比特值。这些比特值就组成了密钥的基础。

量子加密法的先进在于这种方法依赖于量子力学定律。传输的光量子是无法进行窃听的，量子要么被接收者的接收机接收，要么被窃听者接收。因为如果有人进行窃听，窃听动作本身将会对通信系统造成干扰，对通信系统的量子状态造成不可挽回的变化，同时通信双方就会得知有人进行窃听，从而结束通信，生成新的密钥。

在光纤上进行的试验证明，这种加密方法在卫星通信中也是可行的。但是量子加密法有很高的维护要求，普通的铜缆（这种媒介只以电子方式而不是量子方式传送信号）无法使用这种技术，而在宽带光纤通信中就可以进行量子密钥的发送。

在一些实际应用中，通过光纤的量子信息的密钥可以用于加密普通宽带数据信道所传送的信息。这种加密技术在不久的将来就会商业化，但是目前相当长的一段时间内，公钥加密法还是首选。因为这种加密方法可以用于现有的电子化的网络和系统中，而不需要使用光纤，并且公钥加密方法可以用于数字签名，量子加密法则无法做到。

量子信息技术如果能走向实用，那么现代的公钥加密体制将被取代。用公钥加密方法进行加密的信息将会立刻变成明文，所以量子加密技术有比较广泛的未来前景。

2.9　PKI 管理机制

PKI（Public Key Infrastructure）即公开密钥体系。一个成熟的加密体系必然要有一个成熟的密钥管理机制，公钥的管理机制都采用 PKI 机制，它是由公开密钥技术、数字证书、证书发行机构（CA）和相关的公钥安全策略等组成。

PKI 体系建立一套证书发放、管理和使用的体系，来支持和完成网络系统中的身份认证、信息加密、保证数据完整性和抗抵赖性。PKI 体系可以有多种不同的体系结构、实现方法和通信协议。公开密钥算法也就是非对称密钥算法，它的基本原理在前面已经详细介绍。

公钥利用电子证书与其拥有者的姓名、工作单位和邮箱地址等捆绑在一起，由权威机构（Certificate Authority，CA）认证、发放和管理。把证书交给对方时就把自己的公钥传送给了对方。证书也可以存放在一个公开的地方，让别人能够方便地找到。

公共密钥方法还提供了进行数字签名的办法：签字方首先要对发送的数据提取摘要并

用自己的私钥对其进行加密；接收方验证签字方证书的有效性和身份，用签字方公钥进行解密和验证，确认被签字信息的完整性和抗抵赖性。

公共密钥方法通常结合使用对称密钥（单密钥）方法，由计算效率高的对称密钥方法对文件和数据进行加密。

目前在 Internet 上主要使用 RSA 公共密钥方法，密钥长度用 512 位或 1024 位，它是广泛使用的 SSL/TLS 和 S/MIME 等安全通信协议的基础。

2.9.1 认证机构

认证机构 CA 是证书的签发机构。构建密码服务系统的核心内容是如何实现密钥管理。公钥体制涉及一对密钥，即私钥和公钥。私钥只由持有者秘密掌握，无须在网上传送，而公钥是公开的，需要在网上传送，故公钥体制的密钥管理主要是公钥的管理问题。目前较好的解决方案是引进证书机制。

证书是公开密钥体制的一种密钥管理媒介。它是一种权威性的电子文档，形同网络计算环境中的一种身份证，用于证明某一主体（如人、服务器等）的身份及其公开密钥的合法性。在使用公钥体制的网络环境中，必须向公钥的使用者证明公钥的真实合法性。因此，在公钥体制环境中，必须有一个可信的机构来对任何一个主体的公钥进行公证，证明主体的身份及它与公钥的匹配关系。CA 正是这样的机构，它的职责是：验证并标识证书申请者的身份；确保 CA 用于签名证书的非对称密钥的质量；确保整个签证过程的安全，确保签名私钥的安全性；管理证书材料信息（包括公钥证书序列号、CA 标识等）；确定并检查证书的有效期限；确保证书主体标识的唯一性，防止重名；发布并维护作废证书表；对整个证书签发过程做日志记录及向申请人发通知等。

CA 也拥有一个证书（内含公钥），也有自己的私钥，所以它有签字的能力。网上的公众用户通过验证 CA 的签字从而信任 CA，任何人都应该可以得到 CA 的证书（含公钥），用以验证它所签发的证书。

如果用户想得到一份属于自己的证书，他应先向 CA 提出申请。在 CA 判明申请者的身份后，便为他分配一个公钥，并且 CA 将该公钥与申请者的身份信息绑在一起，并为之签字后，便形成证书发给那个用户（申请者）。

如果一个用户想鉴别另一个证书的真伪，就用 CA 的公钥对那个证书上的签字进行验证（如前所述，CA 签字实际上是经过 CA 私钥加密的信息，签字验证的过程还伴随使用 CA 公钥解密的过程），一旦验证通过，该证书就被认为是有效的。CA 除了签发证书之外，它的另一个重要作用是证书和密钥的管理。

认证中心就是一个负责发放和管理数字证书的权威机构。对于一个大型的应用环境，认证中心往往采用一种多层次的分级结构，各级的认证中心类似于各级行政机关，上级认证中心负责签发和管理下级认证中心的证书，最下一级的认证中心直接面向最终用户。处在最高层的是认证中心，它是所有人公认的权威，如人民银行总行的 CA。由于认证机构发放的证书是供各种各样的应用程序用的，因此它必须遵循一定的工业标准，这些标准包括以下几类。

2.9.2　加密标准

数字证书及数字签字实质是信息加密，通用的加密标准如下。

- 对称加密算法：DES、Triple-DES、RC2、RC4 和 CAST 等。
- 非对称加密算法：RSA、DSA 和 Diffie-Hellman 等。
- 散列算法：SHA-1、MD5 等。

2.9.3　证书标准

通用的证书标准是 X.509。

目录服务标准为 X.500 及 LDAP。

2.9.4　数字证书

数字证书是一种能在完全开放系统中使用的证书（例如互联网络），它的用户群绝不是几个人互相信任的小集体。在这个用户群中，从法律角度讲彼此之间都不能轻易信任。所以公钥加密体系采取了另一个办法，将公钥和公钥的主人名字联系在一起，再请一个大家都信得过有信誉的公正、权威机构确认，并加上这个权威机构的签名，这就形成了证书。

由于证书上有权威机构的签字，所以大家都认为证书上的内容是可信任的；又由于证书上有主人的名字等身份信息，别人就很容易地知道公钥的主人是谁。构建密码服务系统的核心内容是如何实现密钥管理。公钥体制涉及一对密钥，即私钥和公钥。私钥只由持有者秘密掌握，无须在网上传送，而公钥是公开的，需要在网上传送，故公钥体制的密钥管理主要是公钥的管理问题。目前较好的解决方案是引进证书机制。

证书是公开密钥体制的一种密钥管理媒介。它是一种权威性的电子文档，形同网络计算环境中的一种身份证，用于证明某一主体（如人、服务器等）的身份及其公开密钥的合法性。在使用公钥体制的网络环境中，必须向公钥的使用者证明公钥的真实合法性。因此，在公钥体制环境中，必须有一个可信的机构来对任何一个主体的公钥进行公证，证明主体的身份及它与公钥的匹配关系。

1. 证书的格式与证书发放

数字证书格式的通用标准是 X.509。证书由 CA 根据 X.509 协议产生，应具备的信息如下。

- 版本号：用来区分 X.509 的不同版本。
- 序列号：由 CA 给予每一个证书的特殊编码。
- 签名算法：用于产生证书所用的方法及一切参数。
- 发出该证书的认证机构：CA 的识别名字。
- 有效期限：包括开始日期和结束日期。

- 主题信息：证书持有人的姓名、服务处所等信息。
- 公钥信息：被证明的公钥值，加上使用这个公钥的方法名称。
- 认证机构的数字签名。

在X.509标准的扩展部分，认证机构可以说明该证书的附加信息，如密钥用途等。

用户想获得认证机构的证书时，首先要向认证机构提出申请，说明自己的身份。认证机构在认真查验用户的身份后，向用户发出相应的数字证书。

认证机构在发放证书时要遵循一定的准则，例如要保证自己发出的证书的序列号没有相同的，没有两个不同的实体获得的证书中的主题内容是一致的，不同主题内容的证书所包含的公开密钥要不相同等。

2. 证书的管理

认证机构应有一证书管理机构来管理它发出的所有证书，证书的管理应通过目录服务来实现。这些管理功能包括：

- 用户能方便地查找各种证书及已经撤销的证书。
  用户在验证发送方数字签字时需要验证用户的身份，这就要检验发送方数字证书。由于该证书可能在其有效期限内被认证机构撤销，用户往往要检查认证机构的已撤销证书，因此能否给用户提供方便的证书查询功能，是认证机构是否成功的重要标准之一。
- 能根据用户请求或其他相关信息撤销用户的证书。
  用户的身份并不是一成不变的。如用户的服务处所改变后，用户的身份也就改变了。这时，原来的证书虽在有效期限内但已无意义，故认证机构就应该根据用户的请求，撤销该证书。
- 能根据证书的有效期限自动地撤销证书。
- 能完成证书数据库的备份工作。

3. 证书库

证书库是证书的集中存放地，是网上的一种公共信息库，用户可从此处获得其他用户的证书和公钥。构造证书库的最佳方法是采用支持LDAP协议的目录系统，用户或相关的应用通过LDAP来访问证书库。系统必须确保证书库的完整性，防止伪造、篡改证书。

4. 密钥备份及恢复系统

如果用户丢失了用于解密数据的密钥，则密文数据将无法被解密，从而造成数据丢失。为避免这种情况的出现，PKI应该提供备份与恢复解密密钥的机制。密钥的备份与恢复应该由可信的机构来完成，例如CA可以充当这一角色。值得强调的是，密钥备份与恢复只能针对解密密钥，签名私钥不能够做备份。

5. 证书作废处理系统

证书作废处理系统是PKI的一个重要组件。同日常生活中的各种证件一样，证书在CA为其签署的有效期以内也可能需要作废。例如，A公司的职员a辞职离开公司，这就

需要终止 a 证书的生命期。为实现这一点，PKI 必须提供作废证书的一系列机制。作废证书有三种策略：作废一个或多个主体的证书；作废由某一对密钥签发的所有证书；作废由某 CA 签发的所有证书。

作废证书一般通过将证书列入作废证书表（CRL）来完成。通常，系统中由 CA 负责创建并维护一张及时更新的 CRL，而由用户在验证证书时负责检查该证书是否在 CRL 之列。CRL 一般存放在目录系统中。证书的作废处理必须在安全及可验证的情况下进行，系统还必须保证 CRL 的完整性。

6．证书的发放政策

证书的发放需要遵照一定的规定和手续进行。证书的发放政策是由发证机构根据自己的服务方式和管理方式来确定的。最简单的情况是在 intranet 上，提供用户的名字、住址、工作证和 E-mail 地址可能就可以了。在复杂的情况下，可能要求用户提供各种证明身份的证件、一些权威性的文件以及提供银行账号，甚至需要亲自到专门的机构去办理申请手续。

7．证书的发放方式

证书的发放方式可以有多种。使用 Net-Pass 智能卡时，证书的发放可以在网上在线签发，也可以进行卡的预制签发。

8．证书的应用

认证机构发放的证书的主要应用有：

- 使用 S/MIME 协议实现安全的电子邮件系统。
- 使用 SSL 协议实现浏览器与 Web 服务器之间的安全通信。
- 使用 SET 协议实现信用卡网上安全支付。

2.10　智　能　卡

使用智能卡方法是当前国际上公认的商业网络安全通信中最好的用户端解决方案。智能卡的外形与普通的信用卡相同，内部有微处理器（CPU）和可重写存储单元（EEPROM），并且有文件管理系统和保护算法。Net-Pass 1.0E 使用目前国内、国际上最高水平的智能卡，其内部有硬件产生密钥和实现的 RSA 加密算法，可以高速完成加密、解密等操作。智能卡是防止篡改的简便方法，它可以向诸如客户身份验证、登录到 Windows 2000 域、代码签名和保护电子邮件之类的任务提供安全性解决方案。

1．使用智能卡的优点

对加密智能卡的支持是微软集成到 Windows 2000 中的公钥基础结构（PKI）的关键功能。使用智能卡具有如下几个其他方法所不具备的独特优点。

- 把用户的重要信息，包括证书、密钥、口令和个人信息等，存放于智能卡中安全保管。

- 加密处理可以在卡内完成，使用于加密的个人密钥等信息不能从卡中读出，从而最大限度地保障通信的安全使用。
- 每张智能卡存放的内容都是独特的，是不可替代的，具有代表使用者身份的意义，提供对操作安全的可管理性。
- 智能卡的拥有者可以方便地携带它，可以到任何地点的连接读卡器的计算机上去完成电子商务操作，不仅安全而且比其他方法更方便。

为了安全地保管私钥和电子证书，在 Windows 2000 中，微软为用户还提供了一套智能卡的基础设施。智能卡因为其高安全性和轻便移动性，势必将发展成为类似鼠标/键盘一样的计算机的标准外设。因此推出了一套基于 32 位 Windows 平台的 Smart Card for Windows 产品，包括 API 和开发工具。众多的智能卡厂家，只要生产符合国际 ISO 工业标准的智能卡产品，就可以在微软公司的 Smart Card 软件平台上操作。

当用 Internet Explorer 向一个认证中心申请电子证书时，就会有一对公钥和私钥自动产生出来。私钥可以存储在智能卡中，公钥和其他身份信息（如姓名、电子邮件地址等）发给认证中心。如果认证中心批准该申请，那么包含公钥的电子证书就会被返回来，存储在智能卡中。这种电子证书的申请过程也可以由管理员设定的批处理方法来进行，用户还可以通过 LDAP 查询 CA 中通信对方的公钥。因为 Windows 2000 的认证服务器是可以与活动目录相结合的，所以这方面的查询很方便。

智能卡存储私钥和电子证书的做法，给最终用户提供了对自己安全信息的最大控制，可以方便地从一台机器携带到另一台机器使用，可以在任何一个地点使用。一般来说，智能卡还会用一个个人密码（PIN）保护起来，在要求高安全性的场合，PIN 可以是一些生物信息，例如指纹等。智能卡中存储的信息是加密的，即使破坏了智能卡也得不到里面的内容。智能卡的读卡器也越来越普遍，有 USB 型的，也有 PC 卡型的，甚至 Windows 终端上也会有智能卡插槽，其逐渐在走向大众化。

通过智能卡登录到网络提供了很强的身份验证方式，因为在验证进入域的用户时，这种方式使用了基于加密的身份验证和所有权证据。

例如，如果某个不怀好意的人得到了用户的密码，就可以用该密码在网络上冒充用户的身份。很多人都选择容易记忆的密码，这会使密码先天脆弱，易受攻击。

在使用智能卡的情况下，不怀好意的人必须获得用户的智能卡和个人识别码（PIN）才能假扮用户。因为需要有另一层信息才能假扮用户，所以该组合明显不易遭受攻击。另一个优点是，连续发生几次不成功的 PIN 输入后，智能卡会被锁定，使得对智能卡进行字典攻击非常困难。

2. 智能卡读卡器

在运行 Windows 2000 的计算机上使用符合即插即用功能的智能卡读卡器。如果使用的是不符合即插即用的智能卡读卡器，那么必须从智能卡读卡器的制造商处直接获取安装说明书（包括相关的设备驱动程序软件）。微软不支持非即插即用的智能卡读卡器。

Windows 2000 支持表 2-1 中所列的智能卡读卡器，它们的驱动程序只在探测到相应的即插即用智能卡读卡器硬件时才被安装。

表 2-1　Windows 2000 支持的需安装的智能卡读卡器

制　造　商	智能卡读取器	接　　口	设备驱动程序
Bull	CP8 Smart TLP3	RS-232	Bulltlp3.sys
Gemplus	GCR410P	RS-232	Gcr410p.sys
Gemplus	GPR400	PCMCIA	Gpr400.sys
Litronic	220P	RS-232	Lit220p.sys
Rainbow	Technologies3531	RS-232	Rnbo3531.sys
SCM	Microsystems SwapSmart	RS-232	Scmstcs.sys
SCM	Microsystems SwapSmart	PCMCIA	Pscr.sys

在安装 Windows 2000 时，对 Gemplus GemSAFE 和 Schlumberger Cryptoflex 智能卡的支持包含在默认安装中，在客户端或服务器上不需要进行任何配置即可使用这些卡。

第 3 章 Windows 2000 操作系统的安全管理

信息安全对今天的网络系统来说，是一个重要而严重的问题，它涉及从硬件到软件、从单机到网络的各个方面的安全性机制。而网络操作系统的安全性是整个网络系统安全体系中的基础环节。Windows 2000 的分布式安全机制，实现了高度的安全性集成，以保护和促进业务的发展。

3.1 Windows 2000 的安全性设计

作为继 Windows NT 之后的新一代的企业级网络操作系统，Windows 2000 的安全特性主要体现在三个方面。

1．对 Internet 上的新型服务的支持

Windows 2000 可以实现移动办公、远程服务、安全通信和基于 SSL/TLS 的电子商务等服务。

2．使用安全性框架

Windows 2000 中有“安全服务提供者接口”，即 SSPI（Security Service Provider Interface），方便了其他验证方式，在 Windows 2000 中对其上层应用层来说，没有任何不同。

3．实现对 Windows NT 4.0 的网络支持

Windows 2000 提供了对 Windows NT 4.0 中采用的 NTLM（NT LAN Manager）安全验证机制的支持。用户可以迁移到 Windows 2000 中，替代 NTLM 的 Kerberos 安全验证机制。

3.2 Windows 2000 中的验证服务架构

在 Windows 2000 中的验证服务是以整个体系框架的结构来完成的，它的具体验证服务架构如图 3-1 所示。

使用安全服务提供者接口 SSPI，Windows 2000 实现了应用协议和底层安全验证协议的分离。不管是 NTLM、Kerberos、SSL 还是 DPA，对于应用层而言，都是一样的。

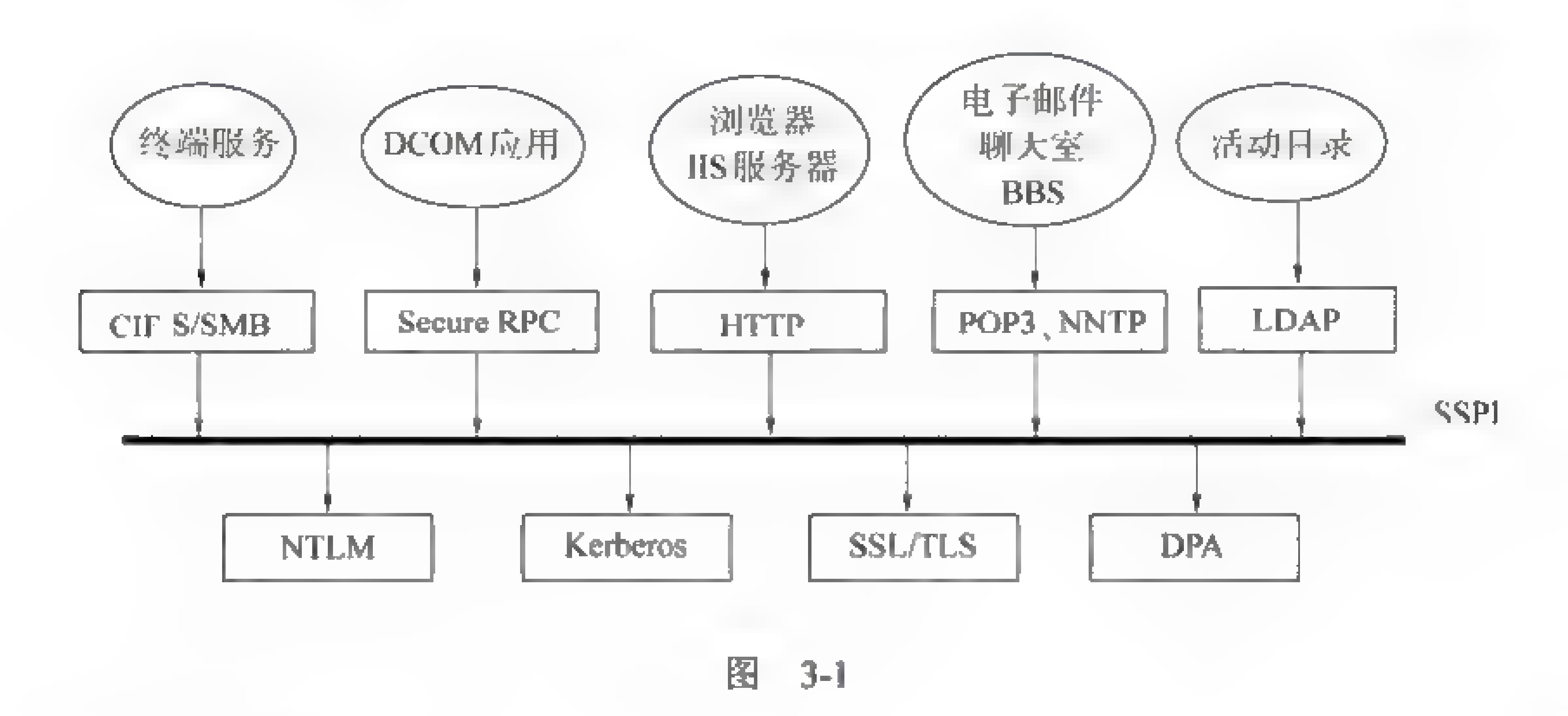


图 3-1

Kerberos 的验证机制

Kerberos 是在 Internet 上广泛采用的一种安全验证机制，它基于公钥技术。Kerberos 协议规定了客户机、密钥发布中心（Key Distribution Center，KDC）、服务器三者之间获得和使用 Kerberos 票证进行通信规则和过程。

Kerberos 验证机制加强了 Windows 2000 的安全性，它使网络应用服务验证速度更快捷，同时可以建立域信任以及在建立域信任的域中传递信任关系，另外 Kerberos 验证有互操作性的优势。在一个多种操作系统并存的异构网络环境中，Kerberos 协议使用一个统一的用户数据库对各种用户进行验证，这样就解决了现在异构环境中的统一验证问题。

3.3 Windows 2000 安全特性

Windows 2000 有数据安全性、企业间通信的安全性、企业和 Internet 的单点安全登录以及易用和良好扩展性的安全管理等安全特性。

1. 数据安全性

数据安全是指数据的保密性和完整性，Windows 2000 的数据安全性如下。

1）用户登录时的安全性

在用户登录网络时，数据的安全保护开始，Windows 2000 使用 Kerberos 和 PKI 验证协议提供了强有力的口令保护和单点登录。

2）网络数据的保护

网络数据指的是本地网络中数据以及不同网络间传送的数据：

- 本地网络的数据是由验证协议以及 IP Security 加密来实现。
- 网络间传送的数据可以通过 IP Security 加密 TCP/IP 通信，Windows 2000 路由和远程访问服务、代理服务等实现。

3）存储数据的保护

存储数据的保密在 Windows 2000 中有文件加密系统以及数字签名等来实现存储数据的保密。

2. 通信的安全性

Windows 2000 提供了多种安全协议和用户模式的内置的集成。它支持虚拟专用网技术以及使用公钥体制，并可以使用电子证书把外部用户映射成为目录服务中的一个用户账户。

3. 单点安全登录

Windows 2000 的用户单点登录网络后，通过网络验证之后，它就可以根据自己拥有的权限访问其相应的服务，Windows 2000 透明地管理一个用户的安全属性（Security Credentials），如图 3-2 及图 3-3 所示。



图 3-2



图 3-3

4. 安全的管理性

Windows 2000 使用安全性模板对计算机进行安全性配置和分析。安全性模板 MMC 提

供多种管理模板从而实现了对工作站、服务器、域控制器的安全管理，在这些安全性模板中，可以配置相应的安全策略。

3.4　Windows 2000 组策略的管理安全

在 Windows 2000 中可以通过组策略来实现安全设置。

3.4.1　Windows 2000 中的组策略

在 Windows 2000 中的组策略有如下几种：

- 账户策略\密码策略。配置密码存留期、长度和复杂性；
- 账户策略\账户锁定策略。配置锁定时间、阈值和复位计数器；
- 账户策略\Kerberos 策略。配置票证寿命；
- 本地策略\审计策略。启用/禁用特定事件的记录；
- 本地策略\用户权限。定义权限，如本地登录、从网络访问等；
- 本地策略\安全选项。修改与注册表值有关的特定安全选项；
- 事件日志。启用成功或失败监视；
- 受限制的组。管理员可控制谁属于特定组；
- 系统服务。控制每个服务的启动模式；
- 注册表。对注册表项配置权限；
- 文件系统。对文件夹、子文件夹和文件配置权限。

1．密码策略

默认情况下，将对域中的所有服务器强制执行一个标准密码策略。密码策略如图3-4所示，默认设置和最低设置如表 3-1 所示。

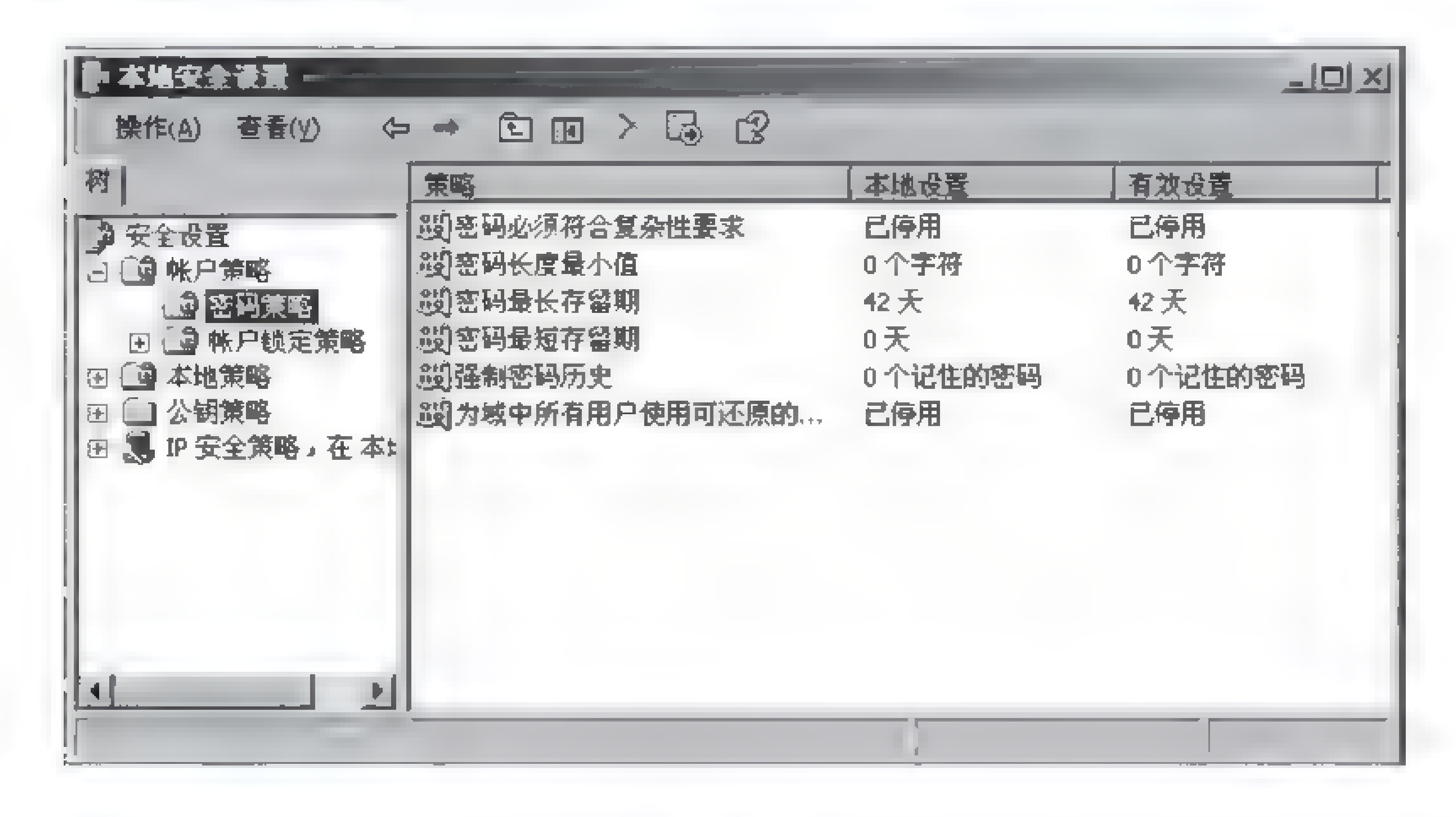


图　3-4

表 3-1 密码策略的默认设置和最低设置

策　　略	默 认 设 置	推荐最低设置
强制执行密码历史记录	记住 1 个密码	记住 24 个密码
密码最长期限	42 天	42 天
密码最短期限	0 天	2 天
最短密码长度	0 个字符	8 个字符
密码必须符合复杂性要求	禁用	启用
为域中所有用户使用可还原的加密来存储密码	禁用	禁用

当组策略的“密码必须符合复杂性要求”设置启用后，它要求密码必须为 6 个字符长（建议将此值设置为 8 个字符）。它还要求密码中必须包含下面类别中至少三个类别的字符：

- 英语大写字母 A，B，C，…，Z。
- 英语小写字母 a，b，c，…，z。
- 西方阿拉伯数字 0，1，2，…，9。
- 非字母数字字符，如标点符号等。

注意： 密码策略不应只对运行 Windows 2000 的服务器强制执行，还应对其他任何要求使用密码进行身份验证的设备强制执行。网络设备，例如路由器和交换机，如果使用简单密码，则极易受到攻击，攻击者可能会尝试控制这些网络设备以便绕过防火墙。

2. 账户锁定策略

有效的账户锁定策略有助于防止攻击者猜出用户的账户的密码，如图 3-5 所示。表 3-2 列出了一个默认账户锁定策略的设置以及针对用户的环境推荐的最低设置。

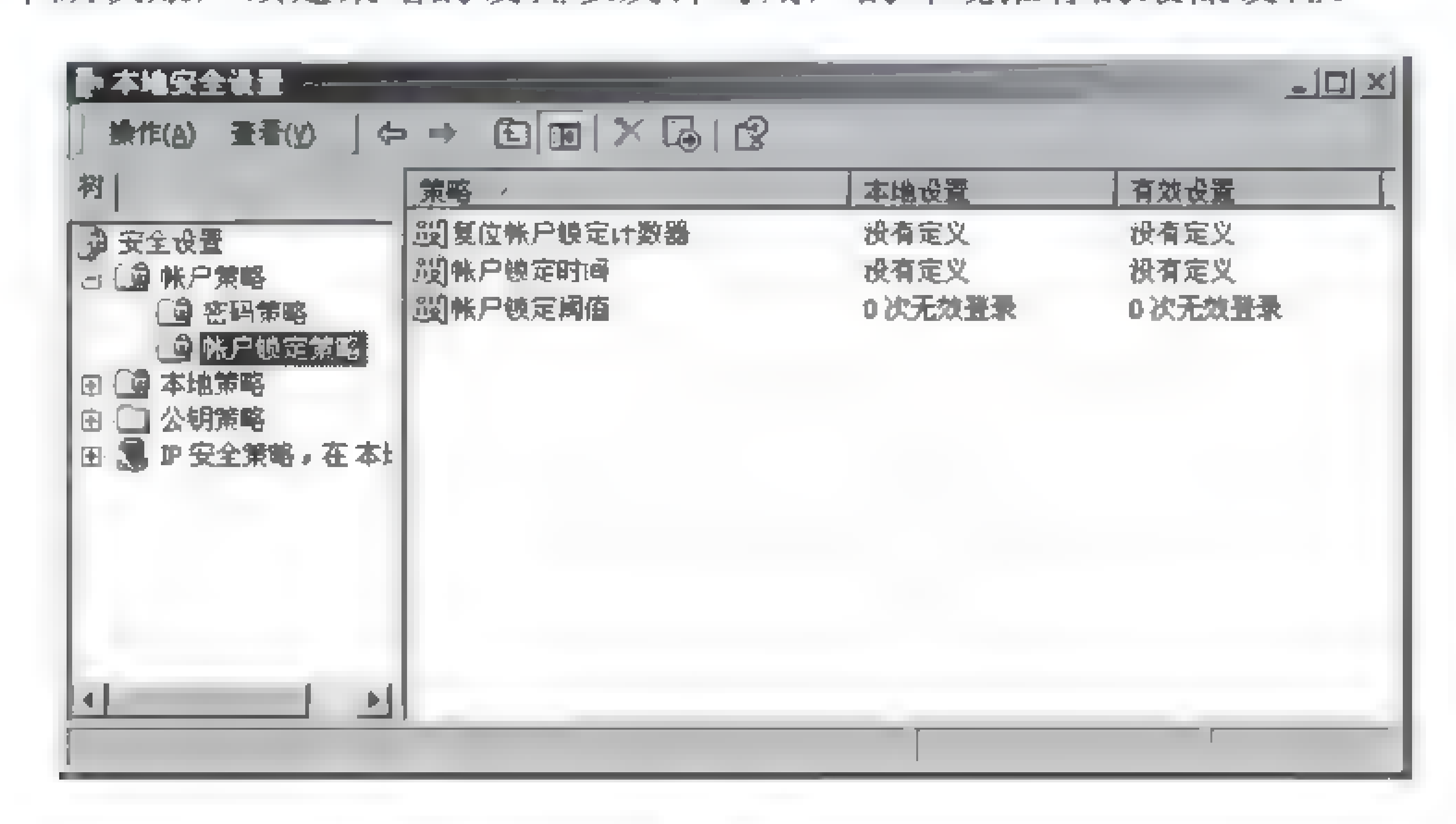


图 3-5

使用表 3-2 中列出的推荐最低设置，若在 30 分钟内经过 5 次无效登录尝试的账户将被锁定 30 分钟（过此时间段后它将复位到 0 次无效登录尝试，于是就可以再次尝试登录了）。只有在管理员将锁定计数复位后才能在 30 分钟之内激活该账户。为提高组织中的安全级

别，你应考虑提高账户锁定期限并降低账户锁定阈值。

表 3-2　默认账户锁定策略设置及最低设置

策　略	默 认 设 置	推荐最低设置
账户锁定时间	未定义	30 分钟
账户锁定阈值	0	5 次无效登录
复位账户锁定计数器	未定义	30 分钟

注意： *密码和账户策略必须在域级别设置。如果在 OU 级别或 Active Directory 中的其他任何位置设置这些账户，它们将影响本地账户而非域账户。*

3. 成员服务器基准策略

一旦配置了域级别的设置，就应该为所有成员服务器定义公用的设置。这是通过“成员服务器 OU（组织单位）”中的一个 GPO（Group Policy Object，组策略对象）完成的，我们称之为基准策略。一个公用的 GPO 可将对各服务器配置特定安全设置的过程自动化。还需要手动应用一些无法通过组策略完成的附加安全设置。

成员服务器的基准组策略如下：

- 审计策略　确定如何在服务器上执行审计。
- 安全选项　使用注册表值确定特定的安全设置。
- 注册表访问控制列表　确定谁可以访问注册表。
- 文件访问控制列表　确定谁可以访问文件系统。
- 服务配置　确定哪些服务需要启动、停止、禁用等。

1）成员服务器基准审计策略

应用程序、安全性和系统事件日志的设置都在该策略中配置并应用到域中的所有成员服务器。各日志的大小都设置为 10 兆字节（MB），而且各日志都配置为不改写事件。所以管理员必须定期查看日志并根据需要进行归档或清理。图 3-6 为成员服务器基准审核策略界面，审核策略的计算机设置见表 3-3。

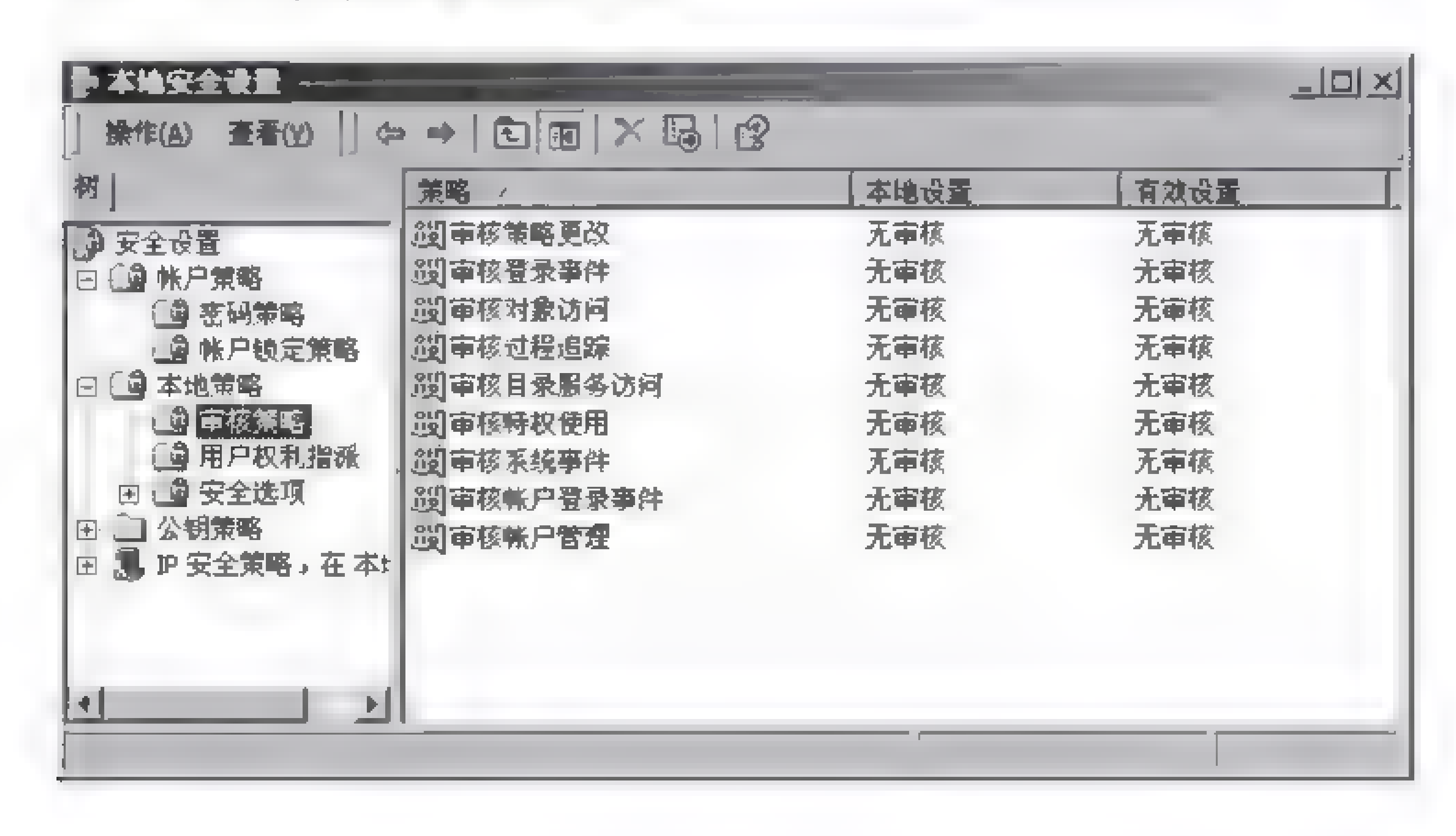


图　3-6

注意： 如果有一个管理系统定期监视日志中的特定事件，将详细信息抽取并转发到一个管理数据库中，就能够捕捉到必需的数据，并因此可以将日志文件设置为在日志满时改写旧事件。

表 3-3　审计策略的计算机设置

策　　略	计算机设置
审计账户登录事件	成功，失败
审计账户管理	成功，失败
审计目录服务访问	失败
审计登录事件	成功，失败
审计对象访问	成功，失败
审计策略更改	成功，失败
审计特权使用	失败
审计过程追踪	无审计
审计系统事件	成功，失败
限制对应用程序日志的来宾访问	启用
限制对安全日志的来宾访问	启用
限制对系统日志的来宾访问	启用
应用程序日志的保留方法	不要改写事件（手动清除日志）
安全日志的保留方法	不要改写事件（手动清除日志）
系统日志的保留方法	不要改写事件（手动清除日志）
安全审计日志满后关闭计算机	未定义

2）成员服务器基准安全选项策略

基准组策略中成员服务器基准安全选项策略设置见表 3-4，用户权限指派设置界面如图 3-7 所示。

表 3-4　基准安全选项策略设置

选　　项	设　　置
对匿名连接的附加限制	没有显式匿名权限就无法访问
允许服务器操作员计划任务	禁用
允许在未登录前系统关机	禁用
允许弹出可移动 NTFS 媒体	管理员
在断开会话之前所需的空闲时间	15 分钟
对全局系统对象的访问进行审计	禁用
对备份和还原权限的使用进行审计	禁用
登录时间过期就自动注销用户	未定义
当登录时间过期就自动注销用户（本地）	启用
在系统关机时清除虚拟内存页面交换文件	启用
对客户端通信使用数字签名（始终）	启用
对客户端通信使用数字签名（如果可能）	启用
对服务器通信使用数字签名（始终）	启用

续表

选　　项	设　　置
对服务器通信进行数字签名（如果可能）	启用
禁用按组合键 Ctrl+Alt+Del 进行登录的设置	禁用
登录屏幕上不要显示上次登录的用户名	启用
LAN Manager 身份验证级别	仅发送 NTLMv2 响应，拒绝 LM&NTLM
用户尝试登录时消息文字	
用户尝试登录时消息标题	
缓冲保存的以前登录次数（在域控制器不可用的情况下）	0 次登录
防止计算机账户密码的系统维护	禁用
防止用户安装打印机驱动程序	启用
在密码到期前提示用户更改密码	14 天
故障恢复控制台：允许自动管理登录	禁用
故障恢复控制台：允许对驱动器和文件夹进行软盘复制和访问	禁用
重命名 Administrator 账户	未定义
重命名 Guest 账户	未定义
只有本地登录的用户才能访问 CD-ROM	启用
只有本地登录的用户才能访问软盘	启用
安全通道：对安全通道数据进行数字加密或签名（始终）	启用
安全通道：需要强（Windows 2000 或以上版本）会话密钥	启用
安全系统磁盘分区（只适于 RISC 操作平台）	未定义
发送未加密的密码以连接到第三方 SMB 服务器	禁用
如果无法记录安全审计则立即关闭系统	启用
未签名驱动程序的安装操作	禁止安装



图　3-7

注意：如果明显增加审计的对象数目，就会有填满安全日志并因而强制系统关闭的风险，于是系统将无法使用，直到管理员清理了日志为止。为防止这一点，应禁用表中所列的关机选项，或者增加安全日志的大小。

在上述安全的配置管理中一定要注意以下几点的管理与配置。

- 对匿名连接的附加限制。

默认情况下，Windows 2000 允许匿名用户执行某些活动，如枚举域账户和网络共享区的名称。这使得攻击者无须用一个用户账户进行身份验证就可以查看远程服务器上的账户和共享名。为了更好地保护匿名访问，可以配置“没有显式匿名权限就无法访问”。这样做的效果是 Everyone（所有人）组将不能匿名访问，也就是对服务器的任何匿名访问都将被禁止，而且对任何资源都要求显式访问。

- LAN Manager 身份验证级别。

Microsoft Windows 9x 和 Windows NT 操作系统不能使用 Kerberos 进行身份验证。可以通过使用 NTLMv2 对 Windows 9x 和 Windows NT 强制执行一个更安全的身份验证协议。对于登录过程，NTLMv2 引入了一个安全的通道来保护身份验证过程。

注意：如果确实要针对 Windows 9x 和 NT 使用 NTLMv2，Windows 2000 客户机和服务器将继续使用 Kerberos 向 Windows 2000 域控制器进行身份验证。

- 对客户/服务器通信使用数字签名。

在高度安全的网络中实现数字签名有助于防止客户机和服务器被模仿（即所谓“会话劫持”或“中间人”攻击）。服务器消息块（SMB）签名既可验证用户身份，又可验证托管数据的服务器的身份。如有任何一方不能通过身份验证，数据传输就不能进行。但在实现了 SMB 后，因为对服务器间的每一个数据包进行了签名和验证，性能最多会降低 15%。

4. 禁用自动运行功能

一个媒体插入一个驱动器，自动运行功能就开始从该驱动器读取数据，这样，程序的安装文件和音频媒体上的声音就可以立即启动。为防止可能有恶意的程序在媒体插入时就启动，组策略禁用了所有驱动器的自动运行功能。

在注册表中 HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\Explorer\ 下的用以禁用所有驱动器上的自动运行功能的设置 NoDriveTypeAutoRun 的 DWORD 为 0xFF。

5. 成员服务器基准文件访问控制列表策略

为加强文件系统安全，应确保对域中的所有成员服务器公用的目录和文件应用限制性更强的权限。“成员服务器基准安全模板”包含了由 hisecws.inf 模板提供的所有文件访问控制列表并添加了许多文件夹和文件的设置。

表 3-5 列出了除由 hisecws.inf 中的设置定义的文件夹外，还有“成员服务器基准策略”保护的其他文件夹。

表 3-5　成员服务器基准策略保护的其他文件夹及应用权限

保护的文件夹	应用的权限
%SystemDrive%\	Administrators：完全控制 System：完全控制 Authenticated Users：读取和执行、列出文件夹内容、读取
%SystemRoot%\Repair	Administrators：完全控制
%SystemRoot%\Security	Creator/Owner：完全控制
%SystemRoot%\Temp	System：完全控制
%SystemRoot%\system32\Config	
%SystemRoot%\system32\Logfiles	
%SystemDrive%\Inetpub	Administrators：完全控制 System：完全控制 Everyone：读取和执行、列出文件夹内容、读取

注意：%SystemRoot%定义了 Windows 系统文件所在的路径和文件夹名，%System-Drive%定义了包含%SystemRoot%的驱动器。

服务器上安装的大量文件中还有许多应进一步锁定。成员服务器基准策略将更改默认 Windows 启动文件中的以及可从命令提示符下运行的许多可执行文件中的 ACL。

3.4.2　加强内置账户的安全

Windows 2000 有几个内置的用户账户，它们不可删除，但可以重命名。我们最熟悉的 Windows 2000 中的两个内置账户是 Guest（来宾）和 Administrator（管理员）。默认情况下，Guest 账户在成员服务器和域控制器上是禁用的，建议不更改此设置。内置的 Administrator 账户应重命名，且其描述也要更改，以防攻击者使用已知用户名破坏一个远程服务器，因为许多有恶意的脚本在攻击服务器时都使用内置的管理员账户进行第一次尝试。

1．加强本地管理员账户安全

每一个成员服务器都有一个本地账户数据库和一个本地管理员账户，此账户对该服务器有完全控制权，所以此账户非常重要。建议重命名此账户，并确保它使用一个复杂的密码。另外还应确保本地管理员密码未在成员服务器间复制。如果它们被复制了，那么获得了对一个成员服务器的访问权的攻击者，将能够访问其他所有使用相同密码的服务器。

不要使本地管理员账户成为域管理组的一部分，因为这样会使它们的能力超出管理成员服务器所需的能力。出于同样的原因，要确保只使用本地账户来管理网络中的成员服务器。

2．加强服务账户安全

Windows 2000 服务一般都在本地系统账户下运行，但它们也可以在一个域用户或本地账户下运行。只要可能，就应使用本地账户而非域用户账户。每个服务都在其服务账户的

安全上下文中运行，所以如果一个攻击者控制了某一成员服务器上的一个服务，则该服务账户可能就会被用来攻击域控制器。在确定使用哪一个账户作为服务账户时，应确保为此账户指定的特权限制在保证此服务成功运行所需的特权范围内，如表 3-6 所示。

表 3-6　各种服务账户的访问权限

在 Windows 2000 计算机上运行服务时的身份验证	仅网内所有 Windows 2000 服务器	多网应用程序，域间有 NTLM 信任关系
本地用户服务账户	无网络资源，仅可在账户的指定特权下进行本地访问	无网络资源，仅可在账户的指定特权下进行本地访问
域用户服务账户	可作为域用户进行网络访问，在用户的特权下进行本地访问	可作为域用户进行网络访问，在用户的特权下进行本地访问
LocalSystem	作为机器账户已验证用户进行网络访问，在 LocalSystem 账户下进行本地访问	没有跨网的网络资源，可在 LocalSystem 账户下进行本地访问

3.4.3　组策略的安全模板

1．组策略的配置设置存储位置

组策略的存储位置是 GPO 位于 Active Directory 中，安全模板文件位于本地文件系统中。对 GPO 所做的更改直接保存在 Active Directory 中，而对安全模板文件所做的更改必须先导回到 Active Directory 内的 GPO 中，才能应用所做的更改。

2．Windows 2000 的安全模板

Windows 2000 的安全模板有如下几种：

- Basicwk.inf 适用于 Windows 2000 Professional。
- Basicsv.inf 适用于 Windows 2000 Server。
- Basicdc.inf 适用于基于 Windows 2000 的域控制器。
- Securedc.inf 和 Hisecdc.inf 适用于域控制器。
- Securews.inf 和 Hisecws.inf 适用于成员服务器和工作站。

注意：Windows 2000 默认安全模板以 inf 文件的格式存储在%SystemRoot%\Security\Templates 文件夹中。

3．安全模板的格式

模板文件是基于文本的文件。表 3-7 列出了策略部分与模板文件部分之间的对应关系。

3.4.4　组策略的实现

为了有效地使用组策略，最好在两个级别应用安全设置：

表 3-7　策略部分与模板文件部分的对应关系

策 略 部 分	模 板 部 分
账户策略	系统访问
审计策略	系统日志 安全日志 应用程序日志
用户权限	特权
安全选项	注册表值
事件日志	事件审计
受限制的组	组成员资格
系统服务	服务常规设置
注册表	注册表项
文件系统	文件安全

- 域级　一般的安全要求，如对所有服务器使用的账户策略和审计策略等。
- OU 级　对特定服务器的安全要求，如 IIS 的服务器等。

图 3-8 为一个组策略的实例。

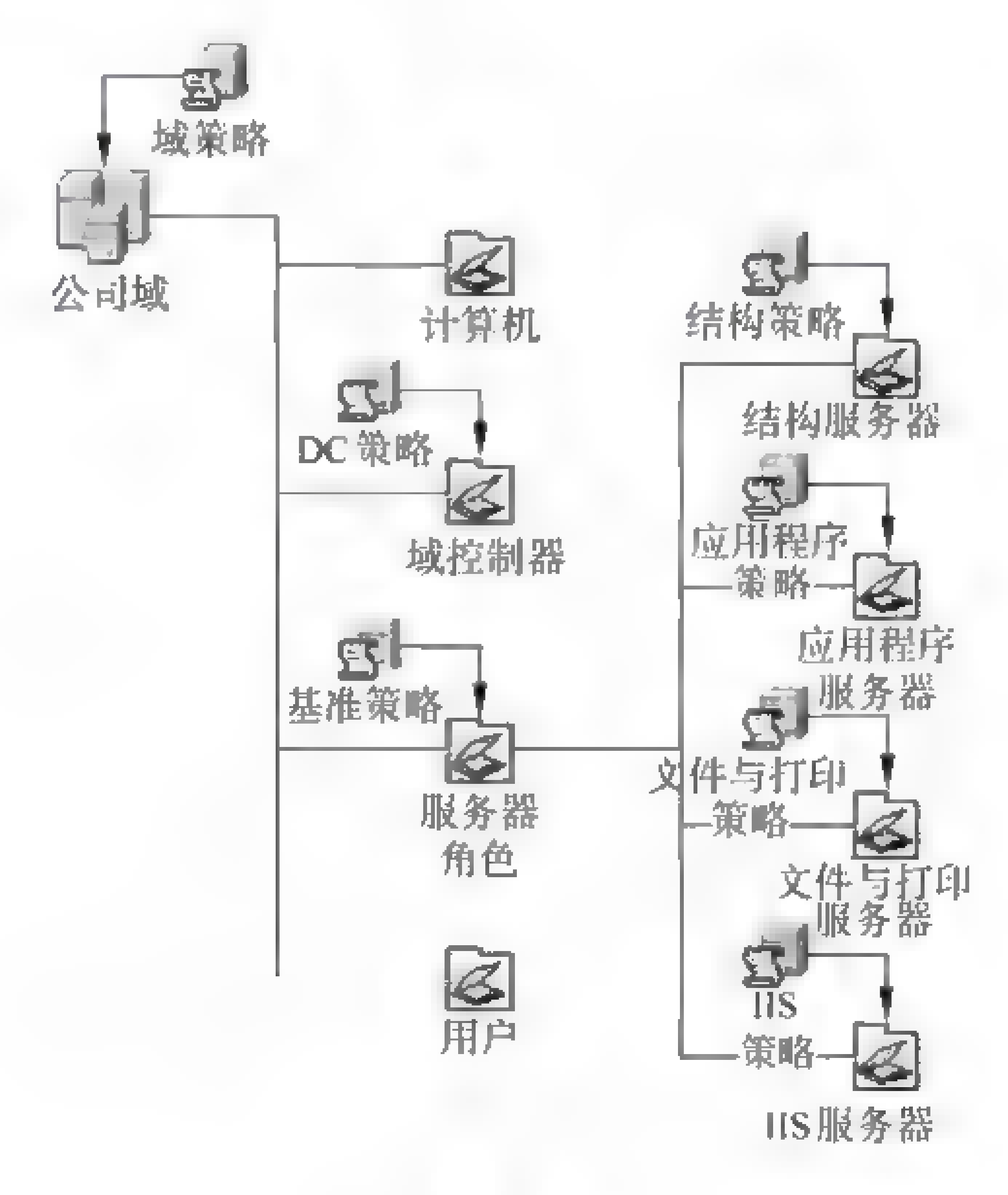


图　3-8

1. 创建 OU 结构

创建 OU 结构的具体步骤如下。

（1）选择“开始”→“程序”→“管理工具”→“Active Directory 用户和计算机”，打开“活动目录”。

（2）右击“域名”，选择“新建”，然后选择“组织单位”选项。

（3）输入成员服务器，然后单击“确定”按钮。
（4）右击成员服务器，选择“新建”，然后选择“组织单位”选项。
（5）输入应用程序服务器，然后单击“确定”按钮。
对文件和打印服务器、IIS服务器和基础结构服务器重复第（5）和第（6）。

2．域级策略

在构建 Windows 2000 域时，创建了一个默认的域策略。对于要应用到整个域中的安全设置来说，可执行以下任一操作。

- 创建另一个策略并将其链接到高于默认策略级别的位置。
- 修改现有的默认策略。

注意：域中一般包含客户机、用户和服务器。建议将服务器安全设置限制在那些必须在域级设置的设置。如果密码和账户策略是在域级设置的，那么它们将只影响域账户；如果这些策略是在 OU 级别或其他任何位置设置的，那么它们只影响本地账户。

1）导入域控制器基准策略
（1）在 Active Directory 用户和计算机中，右击域控制器，然后选择“属性”按钮。
（2）在“组策略”选项卡上，单击“新建”按钮以添加新的组策略对象。
（3）输入 BaselineDC Policy，然后按 Enter 键。
（4）右击 BaselineDC Policy，然后选择“禁止替代”选项。
（5）单击“编辑”按钮，展开 Windows 设置，右击“安全设置”，然后选择“导入策略”选项。
（6）在“策略导入来源”对话框中，浏览 C: \SecurityOps\Templates，然后双击 Baseline-DC.inf。
（7）关闭“组策略”选项卡，然后单击“关闭”按钮。
完成导入域控制器基准策略之后，在域控制器之间强制进行复制，以便所有的域控制器都具有该策略。并在“事件日志”中验证策略是否已成功下载，并验证服务器能否与域中的其他域控制器进行通信。再一次重新启动一个域控制器以确保它能成功地重新启动。
2）导入成员服务器策略
（1）在 Active Directory 用户和计算机中，右击“成员控制器”，然后单击“属性”按钮。
（2）在“组策略”选项卡上，单击“新建”按钮以添加新的组策略对象。
（3）输入 Baseline Policy，然后按 Enter 键。
（4）单击“编辑”按钮，展开 Windows 设置，右击“安全设置”，然后选择“导入策略”选项。
（5）在“策略导入来源”对话框中，浏览 C: \SecurityOps\Templates，然后双击 Baseline.inf。
（6）关闭“组策略”选项卡，然后单击“关闭”按钮。
对于其他应用服务器，它们的相关安全模板如表3-8所示。

表 3-8　其他应用服务器及其安全模板

OU	安全模板
文件和打印服务器	文件和打印 Incremental.inf
IIS 服务器	IIS Incremental.inf
基础结构服务器	基础结构 Incremental.inf

另外要将每个角色所对应的服务器都移到相应的 OU 中，在服务器上使用 secedit 命令来下载策略。并且在“事件日志”中验证策略是否已成功下载，验证服务器能否与域控制器和域中的其他服务器进行通信。在该 OU 中成功测试一个服务器之后，将其余服务器移到该 OU 中，然后应用“安全”，再重新启动每个服务器以确保它们能成功地重新启动。

3.“事件日志”中的事件

如果策略已成功下载，将出现包含以下信息的“事件日志”事件：

- 类型　信息
- 来源 ID　SceCli
- 事件 ID　1704
- 消息字符串　组策略对象中的安全策略被成功应用

在应用该策略之后，可能要过几分钟后才显示此消息。如果没有收到成功的事件日志消息，则需要运行 secedit/refreshpolicy machine_policy/enforce，然后重新启动服务器以强制下载策略。在重新启动之后再次检查“事件日志”以验证策略是否已成功下载。

注意：如果服务在 GPO 中设置为“禁用”且服务器重新启动了一次，则在 GPO 中定义的设置生效之前，这些服务通常已经重新启动。再次重新启动服务器将确保设置为“禁用”的服务不被启动。

4. 策略的验证

策略的验证有以下两种方法。

1）使用“本地安全策略”MMC 验证策略

（1）启动本地安全策略 MMC。

（2）在安全设置下，单击“本地策略”，然后单击“安全”选项。

（3）在右窗格中，查看“有效设置”列。“有效设置”列应当显示在模板中为选定服务器角色配置的设置。

2）使用命令行工具验证策略

secedit

此工具包括在 Windows 2000 中，它可用于显示模板文件和计算机的策略之间的区别。若要将某个模板与计算机上的当前策略进行比较，请使用如下命令行：

```
secedit/analyze/db secedit.sdb/cfg<模板名>
```

注意：如果在应用本指南附带的模板之后运行上述命令，则将产生“访问被拒绝”错误。这是应用了其他安全策略而产生的预期错误，生成的日志文件仍带有分析结果。

3.5 审计与入侵检测

现在随着网络安全技术的快速发展，入侵技术也是日新月异，隐蔽性越来越强，所以为了保证操作系统安全必须有审计和入侵检测。下面介绍审计和入侵检测。

3.5.1 审计

审计的主要目标是识别攻击者对网络所采取的操作。一个攻击者可能企图危害网络上的多台计算机和设备，因此为了了解任何攻击的程度，必须能够协调和合并许多计算机中的信息。

如果用户的日志实用工具已导入到数据库中，那么协调多个日志中的信息就更加容易了。只要所有计算机中的时间是同步的，就可以按时间字段进行排序，并基于时间间隔简化事件跟踪。

审计事件分为两类：成功事件和失败事件。成功事件说明用户成功地获得了访问某种资源的权限，而失败事件则说明用户尝试访问网络的某项资源，但失败了。在 Windows 2000 中的安全事件审计类别一般有八种：登录事件、账户登录事件、对象访问、目录服务访问、特权使用、进程跟踪、系统事件和策略更改。

1. 登录事件

用户每次在计算机上登录或注销时，都会在进行了登录尝试的计算机的安全日志中生成一个事件。另外，在用户连接到远程服务器后，在远程服务器的安全日志中也将生成一个登录事件。在创建或者销毁登录会话和令牌时也会分别创建登录事件。

登录事件对于跟踪以交互方式登录服务器的尝试，或者对于调查从特定计算机发动的攻击十分有用。成功审计将在登录尝试成功的情况下生成一个审计项；失败审计也会在登录尝试失败的情况下生成一个审计项。

1）登录事件 ID

一般在日志中的登录事件的 ID 说明如下。

- 528　用户成功地登录到计算机。
- 529　有人用未知的用户名进行了登录尝试，或者用已知的用户名进行了登录尝试，但密码不正确。
- 530　用户账户试图在不允许的时间进行登录。
- 531　有人使用一个被禁用的账户进行登录尝试。
- 532　有人使用一个过期的账户进行登录尝试。
- 533　未允许该用户登录此计算机。
- 534　该用户试图用不允许使用的登录类型（如网络登录、交互登录、批登录、服务登录或远程交互登录）进行登录。
- 535　指定账户的密码已经过期。
- 536　“网络登录”服务没有处于活动状态。

- 537　由于其他原因登录尝试失败。
- 538　一个用户被注销。
- 539　在有人进行登录尝试时账户被锁定。此事件可表明有人发动密码攻击但未成功，因而导致账户被锁定。
- 540　网络登录成功。此事件表明远程用户从网络成功地连接到服务器上的本地资源，并为该网络用户生成了一个令牌。
- 682　一个用户重新连接到已断开连接的“终端服务”会话。此事件表明有人连接到了以前的“终端服务”会话。
- 683　一个用户没有注销就断开了“终端服务”会话连接。此事件在一个用户通过网络连接到“终端服务”会话的情况下生成，它出现在终端服务器上。

2）利用附录事件中的日志进行安全检测

当事件 ID 出现为 529～534，则表明用户登录失败；如果是猜测用户名和密码，那在日志中会出现 529 和 534，也可能是用户忘记了密码也会出现此 ID 号。如 529 事件后是 528 事件则表明用户的计算机可能已经遭到了密码攻击。当事件 ID 为 530～533 说明用户名和密码正确，但没有登录成功。

2. 账户登录事件

在一个用户登录到域时，是在域控制器上对登录进行处理的。如果审计域控制器上的账户登录事件，那么就会看到在对账户进行验证的域控制器上记录的此登录尝试。账户登录事件是在身份验证程序包对用户的凭据进行验证时创建的。在使用域凭据的情况下，账户登录事件只在域控制器的事件日志中生成。如果出示的凭据是本地 SAM 数据库凭据，那么就会在服务器的安全事件日志中创建账户登录事件。

由于账户登录事件可以记录在域中的任何有效的域控制器上，因此必须确保将各个域控制器上的安全日志合并，以分析域中的所有账户登录事件。

与登录事件一样，账户登录事件也包括计算机登录事件和用户登录事件两种。

1）账户登录事件 ID

作为成员服务器和域控制器基本策略的组成部分，对成功和失败账户登录事件的审计已启用。因此对于网络登录和终端服务身份验证，出现在事件日志中的账户登录事件 ID 说明如下：

- 672　成功地发出并验证了身份验证服务（AS）票证。
- 673　授予了票证授予服务（TGS）票证。
- 674　安全主体更新了 AS 票证或 TGS 票证。
- 675　预先身份验证失败。
- 676　身份验证票证请求失败。
- 677　未授予 TGS 票证。
- 678　账户已成功地映射到域账户。
- 680　识别用于成功的登录尝试的账户。此事件还表明使用身份验证程序包对账户进行了身份验证。
- 681　有人进行了域账户登录尝试。
- 682　一个用户重新连接到已断开连接的“终端服务”会话。

- 683　一个用户没有注销就断开了“终端服务”会话连接。

2）使用账户登录事件 ID 诊断安全事件

对于这些事件中的每个事件，事件日志显示了有关每个特定的登录的详细信息。可以使用账户登录事件项诊断下面的安全事件：

- 域登录尝试失败　事件 ID 675 和 677 表明试图登录到域的失败尝试。
- 时间同步问题　如果客户计算机的时间与进行身份验证的域控制器的时间相差 5 分钟（默认情况下）以上，那么在安全日志中就会记录事件 ID 675。
- 终端服务攻击　可以使“终端服务”会话保持连接状态，以允许进程在会话结束之后继续运行。事件 ID 683 表明用户没有从“终端服务”会话注销，而事件 ID 682 表明有人连接到了先前断开连接的会话。若要防止断开连接或者终止这些已断开连接的会话，请在“终端服务配置”控制台中 RDP-TCP 协议的属性对话框中定义结束已断开的会话的时间间隔。

3. 账户管理

账户管理审计用于确定用户或组是在何时创建、更改或删除的。此审计可用于确定何时创建了安全主体，以及什么人执行了该任务。

1）账户管理事件 ID

作为成员服务器和域控制器基本策略的组成部分，账户管理中的对成功和失败的审计已启用。出现在事件日志中的账户管理事件 ID 说明如下：

- 624　创建了用户账户。
- 625　更改了用户账户类型。
- 626　启用了用户账户。
- 627　尝试了密码更改。
- 628　设置了用户账户密码。
- 629　禁用了用户账户。
- 630　删除了用户账户。
- 631　创建了启用安全的全局组。
- 632　添加了启用安全的全局组成员。
- 633　删除了启用安全的全局组成员。
- 634　删除了启用安全的全局组。
- 635　创建了禁用安全的本地组。
- 636　添加了启用安全的本地组成员。
- 637　删除了启用安全的本地组成员。
- 638　删除了启用安全的本地组。
- 639　更改了启用安全的本地组。
- 641　更改了启用安全的全局组。
- 642　更改了用户账户。
- 643　更改了域策略。
- 644　用户账户被锁定。

2）使用账户管理事件 ID 诊断安全事件

可以使用安全日志项的 ID 诊断下面的账户管理事件。

- 创建用户账户　事件 ID 624 和 626 识别是何时创建和启用用户账户的。如果仅限于为本单位中的特定个人创建账户，那么可以使用这些事件识别是否有未经授权的人员创建了用户账户。
- 更改了用户账户密码　用户本人之外的其他人对密码进行修改，可表明一个账户已经被另一个用户掌握。应查找表明进行了密码更改尝试并获得成功的事件 ID 627 和 628。查看详细信息以确定是否由另一个账户进行了该更改，以及该账户是否为可以重置用户账户密码的服务台或其他服务组的成员。
- 更改了用户账户状态　一个攻击者可能试图通过禁用或删除在发动攻击时使用的账户来掩盖他的踪迹。应该对所有的事件 ID 629 和 630 进行调查以确保这些事件是经授权的事务。还要查找事件 ID 626 后面较短的时间内接着发生事件 ID 629 的情况。这种情况可表明有人启用并使用了被禁用的账户然后又将该账户禁用。
- 对安全组的修改　应该检查对下列组的成员身份进行的更改：域管理员组、管理员组、任一操作员组；自定义全局组、通用组或受到委派承担管理功能的域本地组。对全局组成员身份的修改，请查找事件 ID 632 和 633。对域本地组成员身份的修改，请查找事件 ID 636 和 637。
- 账户锁定　在账户被锁定后，将会在 PDC 模拟器操作主机上记录两个事件。644 事件表明账户名被锁定，然后将记录一个 642 事件，该事件表明用户账户被更改以指示该账户现在已被锁定。此事件只在 PDC 模拟器上记录。

4. 对象访问

可以用系统访问控制列表（SACL）对基于 Windows 2000 的网络中的所有对象启用审计。SACL 包含一个将要审计其对对象进行的操作的用户和组的列表。在 Windows 2000 中用户可以操作的任何对象几乎都有一个 SACL，这些对象包括 NTFS 驱动器上的文件和文件夹，打印机和注册表项。

1）对象访问控制项 ACE

SACL 由访问控制项（ACE）组成。每个 ACE 都包含三部分信息：

- 要对其进行审计的安全主体。
- 要审计的特定访问类型，称为“访问掩码”。
- 指示要审计失败访问、成功访问还是两种访问都审计的一个标志。

如果希望让事件出现在安全日志中，则必须首先启用审计对象访问，然后对每个需要对其进行审计的对象定义 SACL。

2）审计对象访问事件 ID

审计对象访问出现在事件日志中 ID 说明如下：

- 560　授予了对现有的对象的访问权。
- 562　关闭了一个对象的句柄。
- 563　进行了一次打开一个对象以便将它删除的尝试（这在指定了 FILE_DELETE_ON_CLOSE 标志的情况下供文件系统使用）。

- 564　删除了一个受保护的对象。
- 565　授予了对现有的对象类型的访问权。

5．目录服务访问

Active Directory 对象具有与它们关联的 SACL，因此也可以对它们进行审计。通过审计账户管理来审计 Active Directory 用户账户和组账户。要审计对其他名称上下文中的对象的修改，则必须审计对象访问，可以使用 ADSIEDIT MMC 管理单元来修改配置名称上下文的容器和对象的 SACL。完成这项工作的步骤是：在 ADSIEDIT 控制台中显示所需的上下文，然后在高级安全设置对话框中修改对象的 SACL。

由于会激发大量的事件，很难找到目录服务访问的特定事件，因此，对于目录服务访问，成员服务器和域控制器基本策略只审计失败的事件。这将有助于识别一个攻击者试图对 Active Directory 进行未经授权的访问。

尝试的目录访问将在安全日志中显示 ID 为 565 的目录服务事件。只有通过查看安全事件的详细信息才能确定该事件对应于哪一个对象。

6．特权使用

只要用户在网络中，就会行使所规定的用户权限。如果审计"特权使用"的成功和失败，那么每次一个用户尝试行使用户权限时都会生成一个事件。

审计特权使用时并非对所有用户权限进行审计，下列用户权限不在其内：

- 绕过遍历检查。
- 调试程序。
- 创建令牌对象。
- 替换进程级别的令牌。
- 生成安全审计。
- 备份文件和目录。
- 还原文件和目录。

1）特权使用事件 ID

启用了对特权使用的审计出现在事件日志中的特权使用事件 ID 如下：

- 576　向用户的访问令牌中添加了指定的特权（在用户登录时生成此事件）。
- 577　用户试图执行一个特权系统服务操作。
- 578　有人在受保护对象的已打开句柄上使用了特权。

2）通过特权使用事件 ID 诊断安全事件

通过审计特定的用户权限的 ID，可以诊断下面的事件：

- 充当操作系统的一部分　应查找指示了 SeTcbPrivilege 特权的事件 ID 577 或 578。在事件详细信息中标出了使用该用户权限的用户账户。此事件可以表明有一个用户通过充当操作系统的一部分来试图提升安全特权。例如，一个用户试图将其账户添加到管理员组的 GetAdmin 攻击就使用了此特权。此事件的日志项只能属于系统账户以及授予了这一用户权限的任何服务账户。
- 更改系统时间　应查找指示了 SeSystemtimePrivilege 特权的事件 ID 577 或 578。在

事件详细信息中标出了使用该用户权限的用户账户。此事件可以表明有一个用户尝试更改系统时间以隐藏事件发生的真实时间。

- 从远程系统强制关闭　应查找带有用户权限 SeRemoteShutdownPrivilege 的事件 ID 577 和 578。在事件详细信息中会包括给其授予该用户权限的特定安全标识符（SID）和授予了该权限的安全主体的用户名。
- 加载和卸载设备驱动程序　应查找指示了 SeLoadDriverPrivilege 特权的事件 ID 577 或 578。在事件详细信息中标出了使用该用户权限的用户账户。此事件可表明有一个用户试图加载一个设备驱动程序的未经授权的版本或特洛伊木马版本。
- 管理审计和安全日志　应查找指出了 SeSecurityPrivilege 特权的事件 ID 577 或 578。在事件详细信息中标出了使用该用户权限的用户账户。在清除事件日志以及向安全日志写入有关特权使用的事件时都会发生此事件。
- 关闭系统　应查找指出了 SeShutdownPrivilege 特权的事件 ID 577。在事件详细信息中标出了使用该用户权限的用户账户。在有人尝试关闭计算机时会发生此事件。
- 取得文件等的所有权　应查找指出了 SeTakeOwnershipPrivilege 特权的事件 ID 577 或 578。在事件详细信息中标出了使用该用户权限的用户账户。此事件可表明有一个攻击者正在通过取得一个对象的所有权来尝试绕过当前的安全设置。

7. 进程跟踪

如果审计在基于 Windows 2000 的计算机上运行的进程的详细跟踪信息，那么事件日志将显示创建进程和结束进程的尝试。事件日志还会记录一个进程尝试生成一个对象的句柄或尝试获取对一个对象的间接访问权的时间。

由于会产生大量的审计项，因此成员服务器和域控制器基本策略不启用对进程跟踪的审计。选择审计成功和失败的进程跟踪，将会在事件日志中记录下面的事件 ID：

- 592　创建了一个新进程。
- 593　一个进程已退出。
- 594　复制了一个对象的句柄。
- 595　获得了对一个对象的间接访问权。

8. 系统事件

在一个用户或进程改变计算机环境的某些方面时会生成系统事件。可以审计对系统进行更改的尝试，如关闭计算机或更改系统时间。

审计系统事件，则也要审计清除安全日志的时间。这是很重要的，因为攻击者往往试图在对环境进行更改之后清除他们的踪迹。

1）系统事件 ID

成员服务器和域控制器基本策略对成功和失败的系统事件进行审计。

出现在事件日志中的系统事件 ID 如下。

- 512　Windows 正在启动。
- 513　Windows 正在关闭。
- 514　本地安全机构加载了一个身份验证程序包。

- 515　一个受信任的登录进程已向本地安全机构注册。
- 516　为了对安全事件消息进行排队而分配的内部资源已经用尽，导致一些安全事件消息丢失。
- 517　安全日志被清除。
- 518　安全账户管理器加载了一个通知程序包。

2）通过事件 ID 捕获安全方面的信息

- 计算机关闭/重新启动

事件 ID 513 表明 Windows 正在关闭。知道关闭或重新启动服务器的时间是很重要的。有许多合法的原因，例如安装驱动程序或应用程序时需要重新启动，或者在进行维护时关闭或重新启动服务器。不过，攻击者也可能强制服务器重新启动以便在启动过程中获取对系统的访问权。应该将所有的关闭计算机的情况都记录下来，以便与事件日志进行比较。

许多攻击都涉及计算机的重新启动。通过研究事件日志，你可以确定服务器重新启动的时间，以及该重新启动是计划中的重新启动还是未计划的重新启动。事件 ID 512 表明 Windows 正在启动，在系统日志中自动生成的一系列其他事件也表明 Windows 正在启动。这些事件中包括事件 ID 6005，该事件表明启动了事件日志服务。

除了这一日志项外，还应查找在系统日志中是否存在另外两个不同的事件日志项之一。如果前一次关机是完全的，例如在管理员重新启动计算机时，那么在系统日志中会记录事件 ID 6006（事件日志服务已停止）。通过检查该日志项的详细信息，可以确定是哪一个用户进行了该关机操作。

如果重新启动是由意外的重新启动造成的，那么事件 ID 6008（发生在<日期><时间>的前一次系统关闭）是意外的。这也可表明有一个导致计算机关闭的拒绝服务攻击。但是请记住，也有可能是由电源故障或者设备驱动程序故障造成的。

如果重新启动是由蓝屏造成的，那么在系统日志中就会记录一个具有“保存转储”源的事件 ID 1001。可以在事件详细信息中检查实际的蓝屏错误消息。

注意：*若要包括事件 ID 1001 项的记录，必须在“系统控制面板”小程序的恢复设置部分启用将事件写入系统日志复选框的选项。*

- 修改或清除安全日志

攻击者可能试图修改安全日志，或者在实施攻击过程中禁用审计功能，或者清除安全日志以防止被检测到。如果发现安全日志中很多时间段内没有日志项，则应查找事件 ID 612 和 517 以确定哪个用户修改了审计策略，应该将所有的事件 ID 517 与表明清除安全日志的所有时间的物理日志进行比较。一次未经授权的安全日志清除，可能是一次隐藏以前的安全日志中存在的事件的企图（可能是一次隐藏的攻击）。在事件详细信息中包括了清除该日志的用户的名称。

9. 策略更改

审计策略应定义将审计对网络中的修改，它有助于确定是否有攻击你的网络的企图。攻击者会设法修改审计策略本身，以使他们进行的任何更改不会被审计到。

1）策略更改事件 ID

如果网络中的审计策略更改，你会发现修改审计策略以及对其他策略和用户权限的更改。成员服务器和域控制器基本策略对成功和失败的审计策略更改进行审计。出现在事件日志中的策略更改事件 ID 如下：

- 608　授予了用户权限。
- 609　删除了用户权限。
- 610　与另一个域建立了信任关系。
- 611　删除了与另一个域的信任关系。
- 612　更改了审计策略。
- 768　在一个目录林（比目录树更大的域）中的命名空间元素和另一个目录林中的命名空间元素之间检测到了冲突（在一个目录林中的命名空间元素与另一个目录林中的命名空间元素重叠时发生）。

2）使用策略更改事件 ID 检测事件

通过上述的 ID 能检测发生的如下事件：

- 充当操作系统的一部分。应在事件详细信息中查找带有用户权限 SeTcbPrivilege 的事件 ID 608 和 609。
- 将工作站添加到域。在事件详细信息中查找带有用户权限 SeMachine Account Privilege 的事件。
- 备份文件和目录。应在事件详细信息中查找带有用户权限 SeBackup Privilege 的事件。
- 绕过遍历检查。应在事件详细信息中查找带有用户权限 SeChange Notify Privilege 的事件。此用户权限允许用户即使在没有访问目录树的其他权限的情况下遍历该目录树。
- 更改系统时间。应在事件详细信息中查找带有用户权限 SeSystem time Privilege 的事件。此用户权限允许一个安全主体更改系统时间，潜在地掩盖事件发生的时间。
- 创建永久共享对象。应在事件详细信息中查找带有用户权限 SeCreate Permanent Privilege 的事件。此用户权限的拥有者可以创建文件和打印共享。
- 调试程序。应在事件详细信息中查找带有用户权限 SeDebug Privilege 的事件。此用户权限的拥有者可以附加到任何进程；默认情况下，只将此权限授予管理员。
- 从远程系统强制关闭。应在事件详细信息中查找带有用户权限 SeRemote Shutdown Privilege 的事件。
- 增加调度优先级。应在事件详细信息中查找带有用户权限 SeIncrease Base Priority Privilege 的事件。具有此权限的用户可以修改进程优先级。
- 加载卸载设备驱动程序。应在事件详细信息中查找带有用户权限 SeLoad Driver Privilege 的事件。具有此用户权限的用户可以加载设备驱动程序的特洛伊木马版本。
- 管理审计和安全日志。应在事件详细信息中查找带有用户权限 SeSecurity Privilege 的事件。具有此用户权限的用户可以查看和清除安全日志。
- 替换进程级别的令牌。应在事件详细信息中查找带有用户权限 SeAssign Primary Token Privilege 的事件。具有此用户权限的用户可以更改与已启动的子进程关联的

默认令牌。

- 还原文件和目录。应在事件详细信息中查找带有用户权限 SeRestore Privilege 的事件。
- 关闭系统。应在事件详细信息中查找带有用户权限 SeShutdown Privilege 的事件。具有此用户权限的用户可以关闭系统，以初始化新设备驱动程序的安装。
- 取得文件等的所有权。应在事件详细信息中查找带有用户权限 SeTake Ownership Privilege 的事件。具有此用户权限的用户可以通过取得对象或文件的所有权来访问 NTFS 磁盘上的任何对象或文件

10. 第三方应用程序

有多个第三方应用程序可实现本地日志记录功能，以提供有关该应用程序的详细信息。维护日志文件的所有计算机都应使用同步时钟，这样可以使管理员比较计算机之间以及与服务之间的事件，以确定哪些操作是由攻击者进行的。

许多针对计算机的攻击都是这样实现的：攻击安装在目标计算机上的服务，或者将有效的驱动程序替换为包含特洛伊木马的驱动程序版本，以给予攻击者访问目标计算机的权限。

下面的工具可用于监视已安装在计算机上的服务和驱动程序。

1）服务控制台

服务 MMC 控制台用于监视本地计算机或远程计算机的服务，并允许管理员配置、暂停、停止、启动和重新启动所有已安装的服务。可使用此控制台确定是否存在已配置为自动启动的服务当前未启动的情况。

2）Netsvc.exe

此命令行工具包含在 Windows 2000 Server Resource Kit（Windows 2000 Server 资源工具包）中，它使管理员能够从命令行远程启动、停止、暂停、继续服务和查询服务的状态。

3）SvcMon.exe

此工具监视本地和远程计算机上服务的状态变化（启动或停止）。为检测这些变化，Service Monitoring Tool 实现一种轮询系统。在受监视的服务停止或启动时，Service Monitoring Tool 会通过电子邮件通知你。你必须使用 Service Monitor Configuration Tool（smconfig.exe）来配置服务器、轮询间隔和要监视的服务。

4）Drivers.exe

此工具可显示在安装有该工具的计算机上安装的所有设备驱动程序。该工具的输出包括驱动程序的文件名、磁盘上的驱动程序的大小以及链接该驱动程序的日期等信息。链接日期可用于识别任何新安装的驱动程序。如果更新后的驱动程序不是最近安装的，则可以表明它是被替换的驱动程序。应始终将此信息与“事件查看器”中的系统重新启动事件相关联。

3.5.2 入侵检测

入侵检测分为主动检测和被动检测，被动检测就是被攻击后使用检查日志的方法，而

主动检测是有目标地查找攻击并阻止这些攻击。有端口扫描、事件查看器、转储日志工具、EventCombMT 工具等方法，扫描端口是最常用的入侵检测方法之一。

1．扫描端口

Netstat.exe 是一种命令行实用工具，可以显示 TCP 和 UDP 中所有打开的端口。Netstat 命令的语法格式如下：

```
Netstat [-a][-e][-n][-s][-p协议][-r][间隔]
```

- -a　显示所有的连接和监听端口。
- -e　显示以太网统计信息。它可以与-s 选项结合使用。
- -n　以数字形式显示地址和端口号。
- -p 协议　显示由指定的协议的连接，可以是 TCP 或 UDP。如果与-s 选项一起使用以显示每个协议的统计信息，则可以是 TCP、UDP 或 IP。
- -r　显示路由表。
- -s　显示每个协议的统计信息。默认情况下，将显示 TCP、UDP 和 IP 的统计信息，可以使用-p 选项以指定默认值的子集。
- 间隔　重新显示所选择的统计信息，在每次显示之间按间隔设置的秒数暂停。Ctrl+C 组合键可停止重新显示统计信息。如果省略此参数，Netstat 将只打印一次当前配置信息。

2．通过事件查看器中的筛选器

1）被动式检测方法

被动式侵入检测系统涉及对事件日志和应用程序日志进行手动检查，检查涉及分析和检测事件日志数据中的攻击模式。有多种工具、实用工具和应用程序可以帮助检查事件日志。本节概述了如何使用每一种工具来协调信息。

2）事件查看器

Windows 2000 安全日志可以使用事件查看器 MMC 控制台来查看。事件查看器可以用来查看应用程序日志、安全日志和系统日志，可以在“事件查看器”中定义筛选器以查找特定的事件。

3）在事件查看器中定义筛选器

（1）在控制台树中选择特定的事件日志。

（2）从视图菜单中选择筛选器。

（3）选择用于筛选的参数，在“属性”对话框的“筛选器”选项卡上，可以定义下列属性以筛选事件项。

- 事件类型。可以将该筛选器限制为用于信息、警告、错误、成功审计、失败审计这些事件类型或其任意组合。
- 事件源。生成该事件的特定服务或驱动程序。
- 类别。可以将该筛选器限制为用于特定的事件类别。
- 事件 ID。如果知道特定事件 ID，筛选器可以将列表限制为该特定事件 ID。

- 用户。可以将事件显示限制为由特定用户生成的事件。
- 计算机。可以将事件显示限制为由特定计算机生成的事件。
- 日期间隔。可以将显示限制为在特定的开始日期和结束日期之间发生的事件。

在应用该筛选器之后，可以将筛选出的事件列表导出到逗号分隔列表或制表符分隔的列表中，以便导入到数据库应用程序中。

3. Dump Event Log Tool（转储事件日志工具）(dumpel.exe)

Dump Event Log（转储事件日志）是一种命令行工具，它包含在 Windows 2000 Server Resource Kit Supplement One（Windows 2000 Server 资源工具包第一增补版）中，它可将本地或远程系统的事件日志转储到一个制表符分隔的文本文件中，然后将此文件导入到电子表格或数据库中以便进行进一步的研究。该工具还可以用于筛选或过滤某些事件类型。

dumpel.exe 工具使用下面的语法：

```
dumpel -f文件名 [-s \\服务器] [-l日志 [-m源]] [-e n1 n2 n3...] [-r] [-t] [-d x]
```

各参数的意义如下。

- -f 文件名。指定输出文件的文件名。-f 没有默认值，因此必须指定文件。
- -s \\服务器。指定要为其转储事件日志的服务器。服务器名的前导反斜杠是可选的。
- -l 日志。指定要转储哪一种日志（系统日志、应用程序日志还是安全日志）如果指定了无效日志名，则会转储应用程序日志。
- -m 源。指定在哪个源中（如重定向器 rdr、串口等）转储记录。只能提供一个源。如果未使用此开关，则会转储所有事件。如果使用了未在注册表中注册的源，则会在应用程序日志中搜索这种类型的记录。
- -e n1 n2。事件 ID nn 的筛选器（最多可以指定 10 个）如果未使用-r 开关，则只转储这些类型的记录；如果使用了-r，则转储除这些类型的记录以外的所有记录。如果未使用此开关，则会选中指定的源名称中的所有事件。使用此开关时必须同时使用-m 开关。
- -r。指定是要筛选特定的源或记录还是将它们过滤掉。
- -t。指定各个字符串由制表符分隔。如未使用-t，则字符串由空格分隔。
- -d x。转储在过去的 x 天内发生的事件。

4. ISA Server 的侵入检测功能

ISA Server 包含了一个集成的侵入检测系统，该系统可以判断对网络进行攻击的企图并以一组预配置的操作或警告作出响应。为检测出有害的侵入，ISA Server 将网络通信量和日志项与已知的攻击方法进行比较。若有可疑的活动则触发警告，这些警告会使 ISA Server 执行许多操作，可能的操作包括：运行一个程序、发送一封电子邮件、将事件记录在 Windows 事件日志中、停止和启动 ISA Server 服务或这些操作的组合。

在启用了侵入检测时，可以对下面的攻击配置警告。

1）所有端口扫描

攻击者在判断目标计算机或网络上打开的端口时所使用的一种方法。侵入检测引擎将

检测连接到多个端口的多次尝试，并在连接尝试的次数大于管理员配置的阈值时发出一个警告。还可以配置 ISA Server 以便只在已知的端口（1～2048）上检测端口扫描。

2）IP 半扫描

此攻击类似于“所有端口扫描”，但它利用了 TCP 通信是一个三步骤过程这一事实。“IP 半扫描”攻击不发送第三个数据包“TCP 三向握手”以避免被检测到。

3）陆地攻击

将向计算机发送一个数据包，它带有欺骗性源 IP 地址，还带有与目标地址和端口的端口号匹配的端口号。欺骗性数据包导致目标计算机进入一个循环，最终导致计算机崩溃。

4）死亡之 Ping

此攻击涉及向一台计算机发送大量异常大的 ICMP 回应请求（Ping）数据包。目标计算机试图响应所有的数据包，导致缓冲区溢出，从而使计算机崩溃。

5）UDP 炸弹

以某些字段中的非法值构造的 UDP 数据包将导致一些较旧的操作系统在收到该数据包时崩溃。如果目标计算机崩溃，则常常很难确定崩溃的原因。

6）WinNuke

这是一种可用于将 Windows 网络禁用的拒绝服务攻击，称为 WinNuke。一个得逞的攻击可以导致网络连接的中断或脆弱计算机的崩溃。

注意：可以在 ISA Server 管理控制台 Internet Security and Acceleration Server\Servers and Arrays\<服务器名>\Monitoring\Alerts 文件夹中查看侵入企图警告。

3.6 修补程序

因为一个操作系统是由许多软件人员共同编写，虽然经过反复测试，但在发布之后仍然会有漏洞，及时修补漏洞就需要修补程序，因为大量受到黑客攻击的都是那些没有安装最新安全修补程序的系统。在 Windows 2000 中，可以使用 Microsoft 安全工具包，在安全工具中有重要的安全信息最新的 Service Pack，以及针对 Windows NT 4.0、Windows 2000、IIS 和 Internet Explorer 的重要安全修补程序。该工具直接链接到 Windows Update 站点，以确保所有最新的修补程序都能得到安装。在 TechNet 站点上可提供该安全工具包。

此外，有一个修复程序检查器（Hfnetchk）很有用，它是一个命令行实用工具，检查服务器上是否具有所有的安全修补程序，并可以在微软站点下载最新的即时修复程序列表。

命令格式

```
Hfnetchk -[开关]
```

各部分的开关如下：

- -about　所有 Hfnetchk 的信息。
- -h 主机名　指定要扫描的 NetBIOS 机器名，默认为 localhost。
- -fh 主机文件　指定要扫描的 NetBIOS 机器名的文件的名称。
- -i IP 地址　指定要扫描的计算机的 IP 地址。

- -fip IP 文件名　指定要扫描的地址的文件名称。
- -r 范围 指定要扫描的 IP 地址范围。
- -d 域名　指定要扫描的域名。
- -n 网络　扫描本地网络上的所有系统。
- -history 级别　查看历史，正常操作不需要填写。
- -t 线程数　用于执行扫描的线程数量，默认值为 64。
- -o 输出　指定所需的输出格式。
- -x 数据源　指定包含即时修复程序信息的 xml 数据源。默认位置是 Microsoft Web 站点的 mssecure.cab。
- -s 取消标识　取消 NOTE 和 WARNING 消息。则值 1 代表只取消 NOTE 消息，2 代表取消 NOTE 和 WARNING 消息。默认值是显示所有消息。
- -z　不要执行注册表检查。
- -nosum　不要计算文件校验和。校验和测试将计算文件的校验和。
- -b　显示满足最低基本安全要求所需的即时修复程序的状态。
- -v　显示有关 Patch NOT Found、WARNING 和 NOTE 消息的详细信息。
- -f 输出文件名　指定文件的名称以保存结果，默认方式是显示在屏幕上。
- -u 用户名　指定用于登录到远程计算机的可选用户名。
- -p 密码　指定与用户名一起使用的密码。
- -?　显示帮助菜单。



Windows Server 2003 的安全管理

网络操作系统的安全是网络安全的核心，提高网络安全的基点应该是从操作系统的安全入手，Windows 系列服务器目前的最新版本为 Server 2003，微软宣称其为迄今为止微软最强大的 Windows 服务器操作系统。一个安全的网络操作系统的安全性特征是贯穿于整个系统之中的，操作系统要安全就必须保证文件系统、用户账户目录、用户确认系统、存储管理、交换和环境子系统等的安全性。Windows 的安全环境就是将保密性融入每一个组件的创建过程中，Windows Server 2003 的安全设计有很大改进，主要体现在网络身份验证、基于对象的授权、比较完整的安全策略及数据加密保护等，以此来保证服务器的安全。

4.1 Windows Server 2003 安全架构

在 Windows Server 2003 中内置了信任安全架构（Trusted Security Infrastructures，TSI）。下面介绍一下 TSI 架构。TSI 安全架构提供了核心的安全服务，主要包括：

- 身份验证。
- 授权与访问控制。
- 审核与计账。
- 密码管理。
- 安全管理，包括鉴别与安全策略。

从安全架构的观点考虑，信任安全架构引入了新的安全层，即访问层。它是由 Burton Group 组提出来的。Windows Server 2003 内置了 TSI 安全架构，加强了访问层的管理。访问层位于资源层和外围层之间，资源层由各种应用和数据组成。外围层包括一些安全设备，包括防火墙、访问控制路由器、入侵检测系统和虚拟专用网的终端等。

TSI 安全架构如图 4-1 所示。

在信任的架构中，信任主要通过网络操作系统的 4 项安全管理来完成，首先是识别（identification），也就是通常所说的访问控制，然后是验证（authentication），即身份验证，接着是授权（authorization），最后是审核。

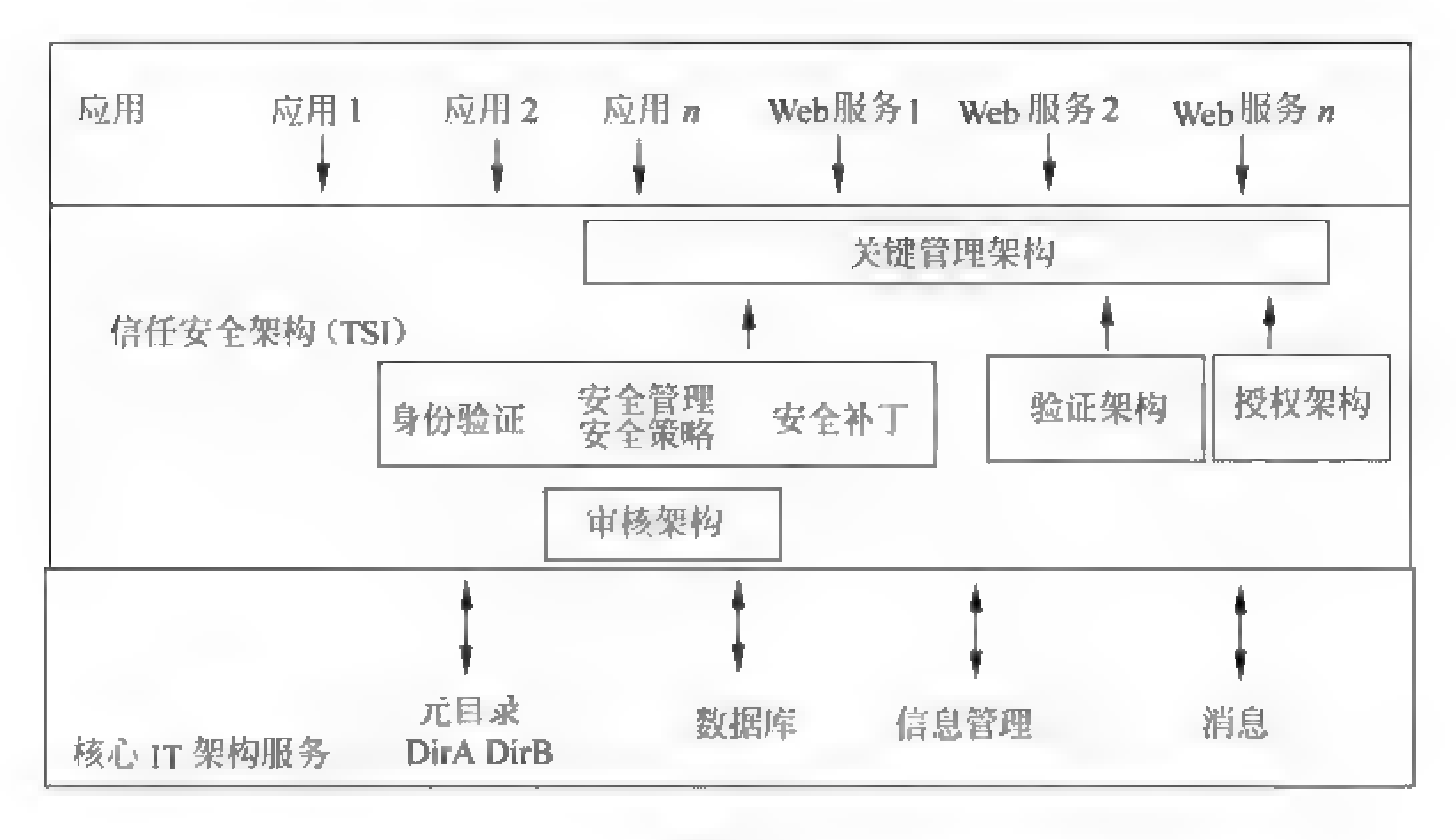


图 4-1

4.2 Windows Server 2003 的新安全机制

Windows Server 2003 的启动速度比 2000 快，系统内新加了许多好用的 DOS 新功能。另外，除具有 2000 的安全机制之外，在 Windows Server 2003 的安全环境中主要进一步加强了用户身份验证和访问控制，即内置了 TSI 架构。在企业服务版本中，还增加了如下的安全功能。

- 授权管理器：Windows Server 2003 中的授权管理器是基于角色安全管理的改进，它可以定义角色及角色执行的任务。还可以使用脚本动态修改权限。
- 存储用户名和密码：Windows Server 2003 的存储用户名和密码功能允许用户连接服务器时使用的用户名和密码与登录网络时使用的用户名和密码不同。此实用工具为用户名和访问 Internet 资源时所需的凭据提供安全存储。
- 软件限制策略：这是 Windows Server 的新安全策略，防止软件应用程序基于软件的哈希算法、软件的相关文件路径、软件发行者的证书或寄宿该软件的 Internet 区域来运行。
- 证书颁发机构：与 Windows 2000 相比，Windows Server 2003 证书服务提供了新的 PKI 功能，旨在展示证书模板编辑功能及为用户和计算机的证书进行自动注册。
- 受限委派：Windows Server 通过这一新的安全功能，可指定要信任的服务用以委派服务器。
- 有效权限工具：此工具将计算授予指定用户或组的权限。
- 加密文件系统（EFS）：在 Windows Server 2003 中不再需要恢复代理。
- Everyone 成员身份：内置 Everyone 组包括 Authenticated Users 和 Guests，但不再包括 Anonymous 组的成员。在以前的 Windows 版本中默认权限是将“完全控制”授予了 Everyone 组，整个文件系统根本没有安全性可言（就本地访问来说）。
- 基于操作的审核：基于操作的审核提供了更多描述性的审核事件，并提供用户选择在审核对象访问时要审核的操作。
- 重新应用安全默认值：此过程可以使用用户重新应用 Windows Server 2003 家族的默

认安全设置。

4.3 Windows Server 2003 的身份验证

身份验证是系统安全的一个基础方面。所有的网络操作系统中的应用程序将被不同的用户进行访问（远程访问或本地访问），操作系统首先必须具有验证能力，才能知道这个用户是不是合法的用户。操作系统将对尝试登录到域或访问网络资源的任何用户进行身份确认。Windows Server 2003 身份验证的重要功能就是它启用对所有网络资源的单一登录。单一登录允许用户使用一个密码或智能卡一次登录到域，然后向域中的任何计算机验证身份。在身份验证方面的增强涵盖了基于本地系统的身份验证和基于活动目录域的身份验证。在本地系统验证方面，默认的设置限制不带密码的本地账户只能用于控制台。这就是说，不带密码的账户将不能再用于远程系统的访问，例如驱动器映射、远程桌面/远程协助连接等。活动目录验证的变化在跨越林的信任方面特别突出。跨越林的信任功能允许在林的根域之间创建基于 Kerberos 的信任关系。在 Windows Server 2003 林中，管理员可创建一个林，将单个林范围外的双向传递性扩展到另外一个 Windows Server 2003 林中。在 Windows Server 2003 林中，这种跨越将两个断开连接的 Windows Server 2003 林链接起来建立单向或双向可传递信任关系。双向林信任用于在两个林中的每个域之间建立可传递的信任关系。

验证一般有以下四种。

- 某个具体内容：如用户名/密码等。
- 某个具体的设备：如 ATM 卡、密钥等，这是一种需要对某个唯一设备进行物理处理以确认用户的验证机制。
- 某种特征：如指纹、视网膜扫描和声音检测等。这是一种可以提供很高安全性的生物验证机制。
- 某个位置：如网络适配卡地址、基于全球卫星定位的系统等。可以提供基于用户位置的验证信息。

操作系统的验证机制一般体现在如下几个方面：

- 支持的验证方法的数量。
- 方法的强度。
- 验证信息是否集成到所有安全操作中。

验证的示意图如图 4-2 所示。

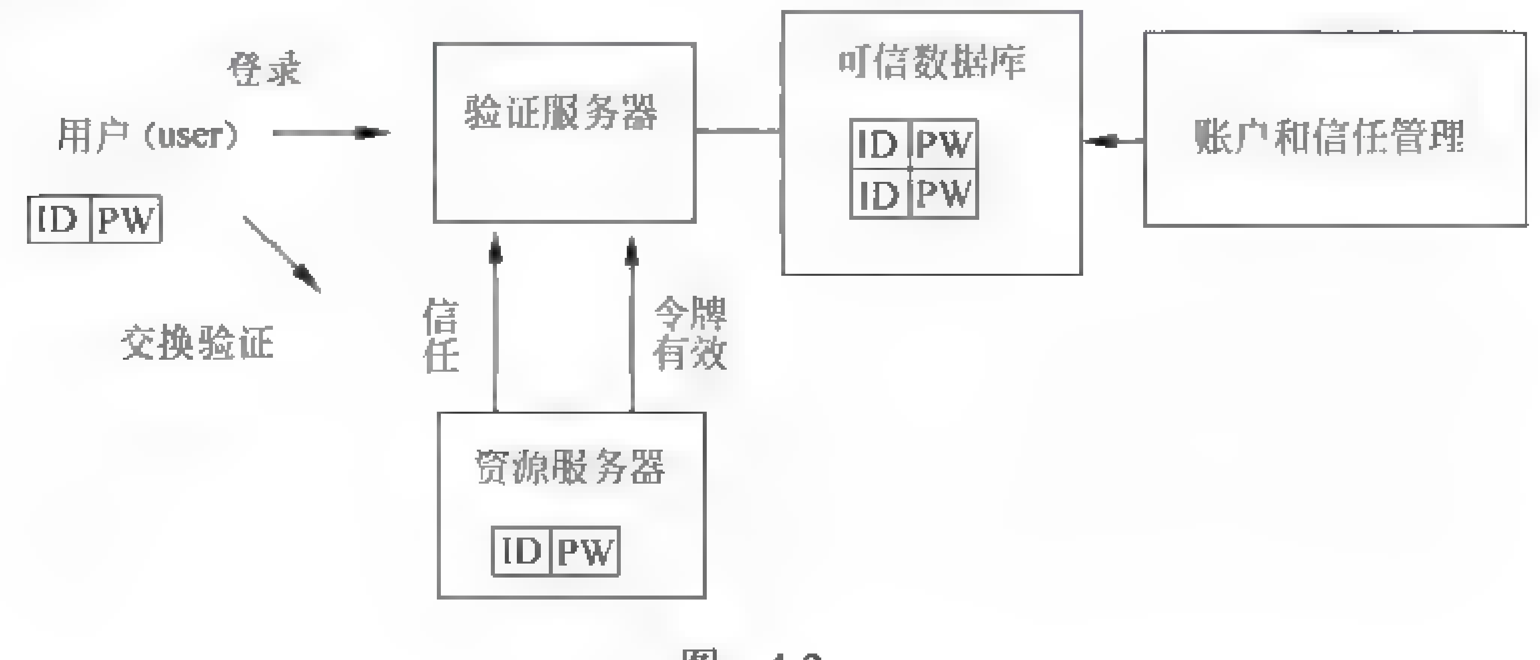


图 4-2

4.3.1 交互验证与网络验证

在 Windows Server 2003 中使用的验证架构与其前版 2000 和 NT 非常相像。即 Windows 在访问系统资源之前进行身份验证。Windows 提供了使用触发安全性的组合键（Ctrl＋Alt＋Del）建立到操作系统的通信，并通过操作系统进行验证。这种信任路径机制的使用可以防止特洛伊木马程序截获一个经过适当培训的用户的初始认证信息。

默认验证机制是用户名和密码，Windows 提供了设置密码有效期限、长度、历史和使用时间的能力。Windows 还针对可猜测性或者字典攻击提供了对管理员密码的强度的密码过滤器。Windows 对储存在注册表中的密码信息进行了加密。这样的手段降低了对密码文件的基于字典的猜测式攻击，无论该文件是在本地机上，还是攻击者把该文件复制到其他机器上。

Windows 除了提供用户名、密码验证外，还提供了标准的图形化标识和验证接口（GraphicalIde Ntificationand Authe NTication，GINA），所以第三方可以很容易地把附加的验证方法集成到 Windows 中。存在广泛的第三方验证机制，从单用途密码到智能卡，从生物测定学方法到多方认证。

除了 GINA，Windows 还提供了安全服务提供者接口（Security Services Provider Interface）和加密应用程序编程接口（Cryptographic Applications Programming Interface，CAPI）。这些模块提供应用程序访问操作系统上的加密服务的标准方法，以及供应商为操作系统提供附加加密服务的标准方法。

Windows 把验证机制集成到全部安全操作和体系中。分布式应用程序使用 Windows 验证机制进行客户/服务器访问。这些验证机制使用挑战/响应协议，这个协议对密码进行数学转换，密码决不会以明文形式传递。结合前面提到的很难被欺骗的信任路径登录，经过认真选择的用户密码是非常强大的。

在 Windows Server 2003 中，不同之处只是增强了相关的安全模块。交互式登录身份验证需要执行两个部分：交互式登录和网络身份验证。成功的用户身份验证取决于这两个过程。下面分别进行介绍。

交互式登录过程向域账户或本地计算机确认用户的身份。这一过程根据用户账户的类型而不同。如果使用域账户，用户可以通过存储在 Active Directory 目录服务中的单一登录凭据使用密码或智能卡登录到网络。如果使用域账户登录，被授权的用户可以访问该域及任何信任域中的资源。如果使用密码登录到域账户，系统将使用 Kerberos V5 进行身份验证。如果使用了智能卡，则需要将 Kerberos V5 身份验证和证书一起使用。

使用本地计算机账户，用户可以通过存储在安全账户管理器（SAM）（也就是本地安全账户数据库）中的凭据登录到本地计算机。任何 1 作站或成员服务器均可以存储本地用户账户，但这些账户只能用于访问该本地计算机。

网络身份验证向用户尝试访问的任何网络服务确认用户的身份证明。为了提供这种类型的身份验证，安全系统支持多种不同的身份验证机制，包括 Kerberos V5、安全套接字层/传输层安全性及为了与 Windows NT 4.0 兼容而提供的 NTLM。

网络身份验证对于使用域账户的用户来说不可见。使用本地计算机账户的用户每次访

问网络资源时，必须提供凭据（如用户名和密码）。通过使用域账户，用户就具有了可用于单一登录的凭据。

4.3.2　Kerberos V5 身份验证

Kerberos V5 是与密码或智能卡一起使用以进行交互登录的协议。它也是 Windows Server 2003 对服务进行网络身份验证的默认方法。在古希腊的神话中，Kerberos 是有三个头的狗，它守卫着地狱的大门。这里使用 Kerberos 就说明这是一个很好的身份验证协议。在 Windows 2000 以后的一系列服务器平台都包括客户端 Kerberos 验证。2000 以前的版本是不包含此验证协议的。Kerberos 的三个头在协议中分别代表验证、授权和审核。但 Kerberos 协议不仅仅具有三个头，而且还具备一些扩展的功能，如密钥的交换等。基本 Kerberos 协议的第 5 版读者请参考 RFC 1510。在协议的三个功能中，Windows Server 2003 中实现了前两个功能，在此小节中只介绍验证。所以在 Windows Server 2003 中，Kerberos 协议也用于数据包验证、提供保密性服务等。

Kerberos V5 身份验证机制颁发用于访问网络服务的票证。这些票证包含加密的数据，其中包括加密的密码，用于向请求的服务确定用户的身份。除了输入密码或智能卡凭据，整个身份验证过程对用户都是不可见的。

Kerberos V5 中的一项重要服务是密钥发行中心（KDC）。KDC 作为 Active Directory 目录服务的一部分，在每个域控制器上运行，它存储了所有客户端密码和其他账户信息。当两个实体都想互相验证对方的身份，如用户和资源服务器之间，这就需要双方都信任的第三方，这就是 Kerberos 的密钥发行中心所完成的任务。

Kerberos 身份验证具有许多优点。

（1）使用独特的票证系统并能提供更快的身份验证。每个需要访问其他域的用户都可以从本地 Kerberos KDC 中被验证。资源服务器将票证作为访问资源服务器的依据，票证可以多次使用并可以存储在客户端。

（2）是交互式验证。

（3）是开放的标准。其 RFC1510 可以通过网址 http://www.ietf.org 下载。

（4）支持委托验证。

（5）支持智能卡网络登录的验证。

下面详细介绍一下 Kerberos 协议。

Kerberos 验证是基于对称密钥加密，其中 Kerberos KDC 提供了票证的认证。Kerberos 票证提供了会话密钥的安全传输，Kerberos KDB 分发会话密钥给需要的客户端。

Kerberos V5 身份验证过程按如下方式工作：

（1）客户端系统上的用户使用密码或智能卡向 KDC 进行身份验证。

（2）KDC 为此客户颁发一个特别的票证授予式票证。客户端系统使用 TGT 访问票证授予服务（TGS），这是域控制器上的 Kerberos V5 身份验证机制的一部分。

（3）TGS 接着向客户颁发服务票证。

（4）客户向请求的网络服务出示服务票证。服务票证向此服务证明用户的身份，同时也向该用户证明服务的身份。

Kerberos V5 服务安装在每个域控制器上，并且 Kerberos 客户端安装在每个工作站和服务器上。每个域控制器作为 KDC 使用。客户端使用域名服务（DNS）定位最近的可用域控制器。域控制器在用户登录会话中作为该用户的首选 KDC 运行。如果首选 KDC 不可用，系统将定位备用的 KDC 来提供身份验证。

对于在安装过程中所有加入到 Windows Server 2003 或 Windows 2000 域的计算机都默认启用 Kerberos V5 身份验证协议。Kerberos 可对域内的资源和驻留在受信任的域中的资源提供单一登录。可通过那些作为账户策略一部分的 Kerberos 安全设置来控制 Kerberos 配置的某些方面。例如，可设置用户的 Kerberos 5 票证生存周期。作为管理员，可以使用默认的 Kerberos 策略，也可以更改它以适应环境的需要。

使用 Kerberos V5 进行成功的身份验证需要两个客户端系统都必须运行 Windows 2000、Windows Server 2003 家族或 Windows XP Professional 操作系统。

如果客户端系统尝试向运行其他操作系统的服务器进行身份验证，则使用 NTLM 协议作为身份验证机制。

下面介绍 Kerberos 身份验证相关组策略的配置。

Windows Server 2003 账户策略包括相关的 Kerberos 策略。

由图 4-3 可知，有 5 项 Kerberos 策略。

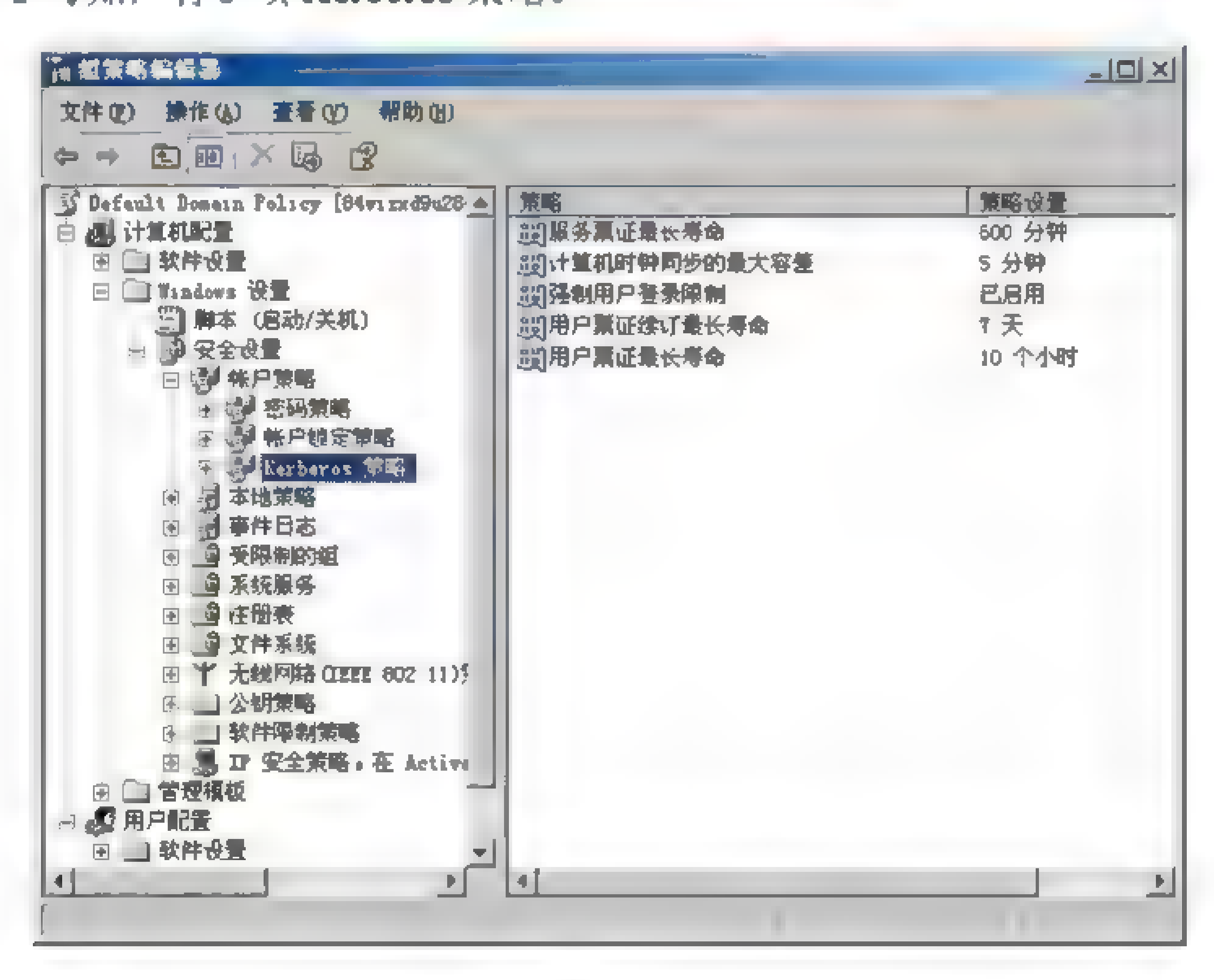


图 4-3

- 服务票证最长寿命：在微软的术语中，服务票证就是 Kerberos 票证，默认的寿命是 10 个小时。
- 计算机时钟同步的最大容差：这是票证时间戳和当前 KDC 时间的最大时间差。Kerberos 使用时间戳针对重放攻击。默认的时间是 5 分钟。
- 强制用户登录限制：这个策略设置是当票证请求提交时，强制 KDC 来验证用户账户的有效性。如果用户没有正确的登录账户或者其账户已不能使用时，用户就不能

获得票证。这个策略默认是启用的。

- 用户票证续订最长寿命：默认时，票证在发行后 7 天仍续订有效。
- 用户票证最长寿命：在微软的术语中，用户票证是 Kerberos TGT。默认的寿命是 10 个小时。

Kerberos 策略只能在域的基础上进行设置。如果其他组织和单元需要使用 Kerberos 策略，可以为域中的其他站点，组织单元的用户和计算机配置委托验证。这也是 Windows 2003 Server 新具备的属性。

4.3.3　存储用户名和密码

首先，在 Windows Server 2003 中的密码发生了改变，使用了强密码。密码在保证网络安全中扮演着非常重要的角色。密码为抵御非法访问构筑了第一道防线。Windows Server 2003 可以在操作系统启动时检查 Administrator 账户密码的复杂程度。如果密码为空或者不满足复杂性要求，将显示 Windows Installer 对话框，警告 Administrator 账户不使用强密码可能存在危险。如果继续使用空密码，用户将无法通过网络访问该账户。

在 Windows Server 2003 中，强密码的规则是：

- 长度至少有七个字符。
- 不包含用户名、真实姓名或公司名称。
- 不包含完整的字典词汇。
- 与先前的密码大不相同。递增密码（Password1、Password2、Password3、…）不能算作强密码。
- 包含全部下列四组字符类型。

<table>
<tr><th>组</th><th>示例</th></tr>
<tr><td>大写字母</td><td>A、B、C、…</td></tr>
<tr><td>小写字母</td><td>a、b、c、…</td></tr>
<tr><td>数字</td><td>0, 1, 2, 3, 4, 5, 6, 7, 8, 9</td></tr>
<tr><td>键盘上的符号（键盘上所有未定义为字母和数字的字符）</td><td>` ~ ! @ # $ % ^ & * ( ) _ + - = { } | [ ] \ :
" ; ' < > ? , . /</td></tr>
</table>

在管理 Windows Server 2003 的所有资源时，均要使用不同的强密码，包括远程资源账户、本地计算机账户和域账户。这样先从密码这一关使 Windows Server 2003 的安全增强。另外，在 Windows Server 2003 中增加了存储用户名和密码的实用工具，它将用户登录的用户名、密码等信息存储为用户配置文件的一部分。这就意味着用户可使用这些用户名和密码遍历网络上的所有计算机。

使用户连接服务器时使用的用户名和密码与登录网络时使用的用户名和密码可以不同。例如，管理员可能使用标准的用户名和密码登录网络，但需要使用可以执行特定任务的管理访问权限连接远程服务器。在这种情况下，该用户必须提供本连接所需的另一个用户名和密码。该用户也可能需要存储该用户名和密码以备将来再使用。

当 Windows Server 2003 的成员要连接到网络上的新计算机时，它会向这台计算机提供

当前用户名和密码。如果所提供的用户名和密码不足以提供访问权限，则“存储用户名和密码”将尝试提供必要的用户名和密码。所有储存的用户名和密码都将接受检查，包括适合资源的最具体到最不具体的用户名和密码，并按这种顺序使用这些用户名和密码逐个进行连接。由于是按照从最具体到最不具体的顺序读取和应用用户名和密码，所以为每个单独的目标或域所存储用户名和密码不会超过一个。

“存储用户名和密码”还允许用户保存为了重新使用而提供的用户名和密码。该信息存储在用户配置文件的安全部分且不能由其他用户访问。如果将用户配置为使用整个企业范围内的唯一配置文件，则无论用户从何处登录到网络都可以使用已存储用户名和密码。

但是这项新功能也会产生很高的危险，所以应该仅在适当的时候存储用户名和密码。

具体的操作是：单击开始菜单，选择控制面板，然后双击存储用户名和密码，打开如图4-4所示的对话框。对此项新功能进行配置。

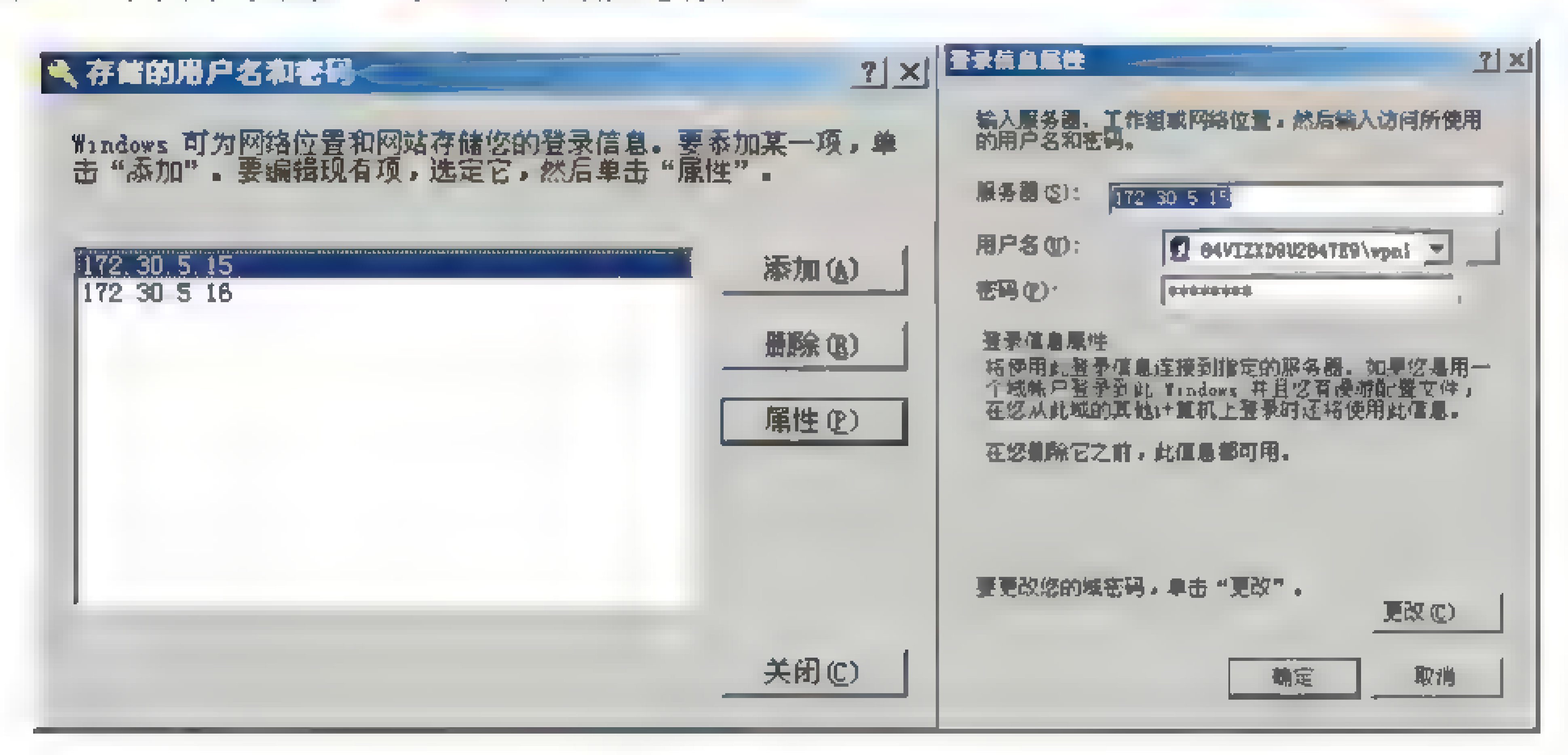


图 4-4

4.4 Windows Server 2003的授权

当用户完成身份验证之后，操作系统需要以某种方式限制此用户对域中计算机资源的访问，在大多数据环境中，用户对域中具体计算机资源的访问都是受限的。所以管理员可以对用户的权利和权限进行管理，而使系统保持安全状态。

4.4.1 授权基础

实际上，访问控制的双方的关系为一方要访问资源，而另一方需要控制用户的访问。在这两者之间，实际上需要第三方。在Windows环境中，第三方称为安全参考监视器（Security Reference Monitor，SRM），SRM是运行于Windows操作系统内核模式中的具有特权的安全组件。它检验所有来自用户端访问资源的请求。授权不仅处理平常所理解的访问控制，如对文件、打印机和注册密码等资源的访问，还能控制对操作系统的进程和线

程的访问。另外授权还可以执行一些相关任务，在微软系统中这些任务称为用户权利。Windows Server 2003 增加了授权的一些新特征，但其基本模式没有发生变化。仍然主要体现在几个重要的概念：访问令牌、访问掩码、安全描述符和用户账户模拟上。图 4-5 显示了 Windows Server 2003 的授权模型。

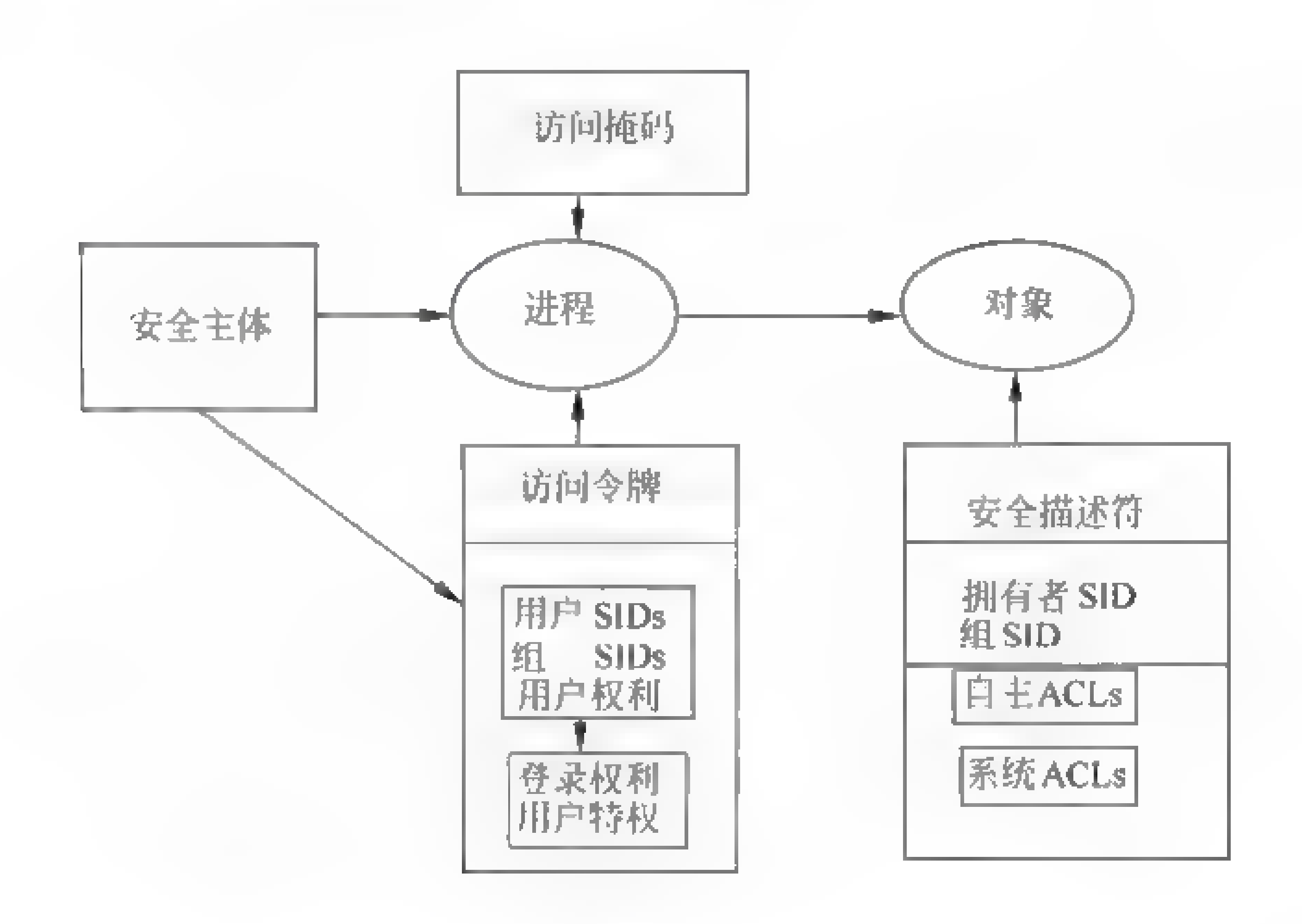


图　4-5

访问令牌与用户登录会话链接，当每个用户登录系统时，产生访问令牌。在操作系统中产生令牌的组件是本地安全认证机构（Local Security Authority，LSA）。令牌包含用户域的授权信息和用户本地授权信息。本地授权信息存储在系统本地安全数据库 SAM 中。它包含了用户本地组成员和本地用户权利。可以使用 whoami 工具来查看令牌的授权信息，如图 4-6 所示。

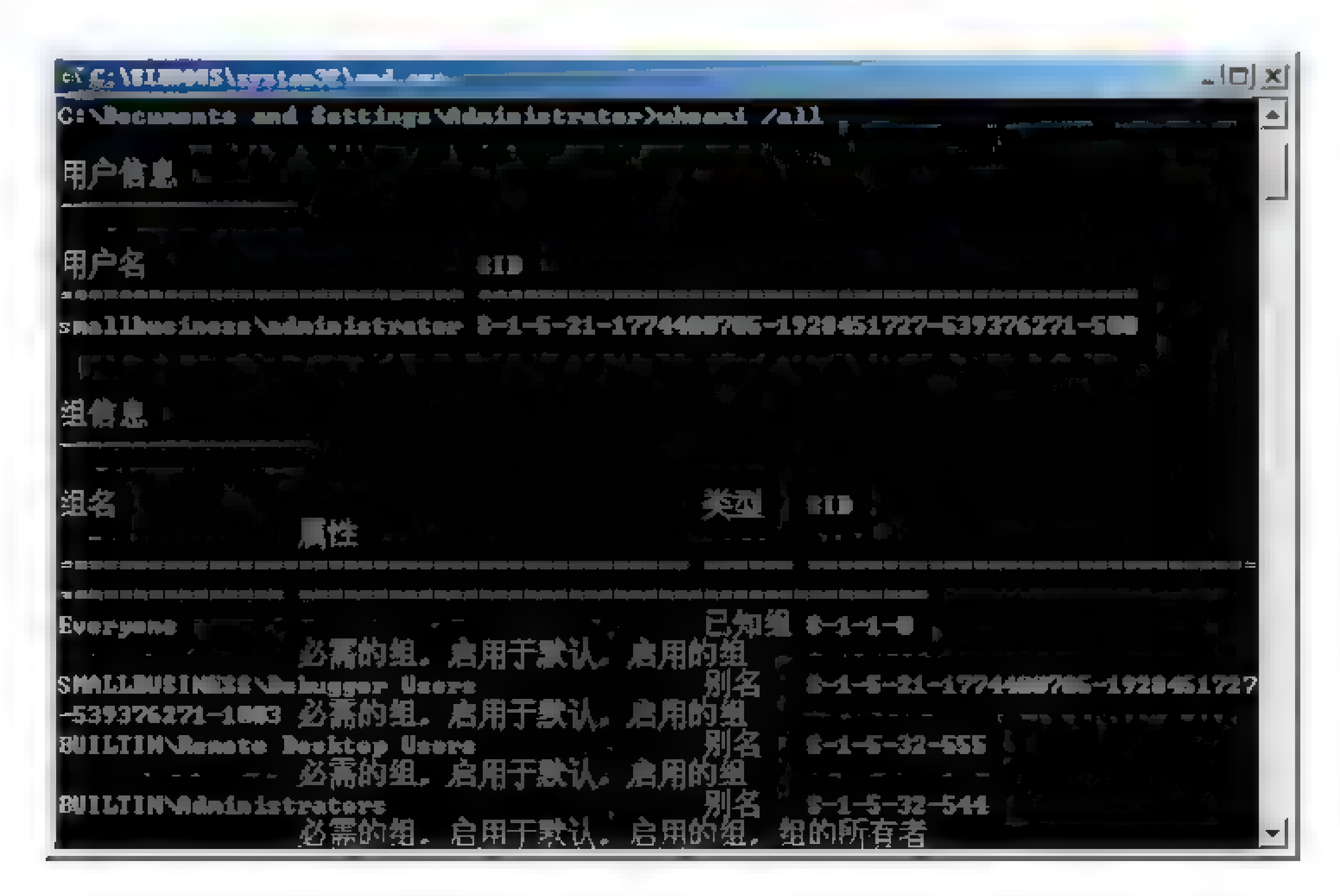


图　4-6

对象方的授权属性称为安全描述符。安全描述符给授权系统提供授权的信息。每个对象都有安全描述符与授权安全对象相链接。安全对象可以在不同用户间共享，而这些用户都可以有不同的安全授权集。安全体可以是文件、文件夹和打印机等任何对象和服务。其中文件系统的安全描述符存储在NTFS文件系统中。

每个安全描述符都包含访问控制列表集（Access Control Lists，ACL）。ACL由多个访问控制实体（Access Control Entries，ACE）组成。ACE与安全标识符（Security Identity，SID）链接来确定用户的访问权限，也就是访问控制，即可以指定读取、修改或者复制特定的系统对象的权限。

访问控制有不同的级别，从字节级别到系统级别，并且可以基于一组系统级别模型。

- 托管访问控制要求进行集中的授权以决定对某人什么是可以访问的，数据所有者和创建者不可以修改访问控制。
- 自由选择访问控制允许对象的所有者定义和修改分配给对象的访问控制。
- 基于角色的访问控制，允许给用户分配角色，然后对角色应用访问规则。这样可简化访问规则的管理并且可以提供更高的一致性。
- 操作系统支持的访问控制可以根据可使用的粒度，用来强化控制的机制强度，以及集成到系统管理机制的程度。

Windows 提供两种形式的访问控制：对象许可和系统范围用户权利。对象许可是自由选择访问控制，对象的创建者可以设置它们。用户权利，在分配给组的时候，允许一种基于角色的访问控制。Windows 的内核包括安全参考监视器，在系统的一个单一模块上实现了这些访问仲裁控制。

在基于Windows的系统上的所有资源，例如文件（只有在NTFS上）、进程、打印机、注册表及通信设备，在对象监视器中都用一个对象来表示。在一个对象上执行指定操作的许可存放在访问控制列表中。ACL提供了一个细粒度的访问控制，它们允许对象所有者基于独立用户，或者它们定义的用户组及管理员定义的组指定许可。为了管理上的方便同时提供了本地组和全局组。关于ACL，很重要的一点是，它们是列表，所有者可以在一个对象上定义的规则的数量是没有限制的。这样，所有者可以指定一个非常全面的列表以完全实现他们希望实现的访问控制。

系统范围用户权利是一种赋予用户能力，或者权限来进行特定系统动作，而不会提供超出他们需要的权限的方法。可以分别管理27种权限。这些权利和限制在登录的时候包含在验证过的用户名字中，而且只能在用户被授予这种权限的时候才能改变（通常保留给网络管理员）。这意味着没有应用程序能为一般的用户修改他们自己的安全性设置，无论是偶然还是病毒侵袭。

下面给出访问控制列表的内容，如图4-7所示。



图 4-7

在安全描述符中，访问权利是由访问掩码十六进制值表示的，如0*1000、0*2000和0*3000分别表示删除、读取和写入的权利，所有的权利总和即0*6000将是用户访问掩码。每个安全描述符都包含两种类型的ACL：自主访问控制列表和系统访问控制列表。自主访问控制列表包括对象所有者设置的ACE。之所以称为自主是由于ACE是由对象所有者设定的。所有者是

Windows 安全模型中的一个关键概念。所有者可以对管理的对象进行授权，同时由于对象的所有者就是 Windows 用户账户，因此由它们来创建对象。这些用户账户可以是域管理员、组管理员等，这将在账户策略中详细讲述。在对象的安全描述符中，所有者由所有者 SID 字段中的 SID 来表示。

系统 ACL 包括对象审核集，是由管理员设定的，它是非自主的。它们与对象的所有者没有任何关系。系统访问控制列表是对象级上的审核。这将在审核策略中详细介绍。

在 Windows 的授权模型中，用户不是自己访问系统资源的，而是由服务器进程来完成用户的访问资源请求。进程是用户账户的模拟。当进程模拟用户时，这意味着用户运行在安全的上下文，它正在使用用户授权的属性。

在 Windows Server 2003 中有以下模拟级：匿名（anonymous）、识别（identify）、模拟（impersonate）和委托（delegate）。

- 匿名：进程模拟匿名用户，访问令牌不包含任何授权信息。
- 识别：进程能使用用户的安全标识来执行相关的安全进程，但不能模拟用户。
- 模拟：进行能代表用户访问本地计算机资源。访问令牌将包含用户授权信息。
- 委托：服务能够代表用户访问本地计算机资源和远程计算机资源。访问令牌将包含用户授权信息。

4.4.2　Windows Server 2003 的授权

在 Windows Server 2003 的授权中主要增加了有效的权限表，默认活动目录（AD）对象安全描述符的改变和 AD 对象配额概念的使用。

1. Windows Server 2003 授权的受限

在 Windows Server 2003 中有许多受限的默认授权集。

（1）NTFS 根目录的权限受限更严，非管理员在此目录中既不能写也不能修改任何由其他用户创建的文件。而 2000 中 everyone 都有完全控制的权限。在 2003 中权限如下。

- Administrator、System Account 和 Creator Owner：完全控制。
- Everyone：读取/执行。
- User：读取/执行，生成文件夹/追加数据，生成文件/写数据。

（2）默认共享权限限制更严。

Everyone 现在只有读取的权限。这就是说对于新创建的共享，即使是 Administrator，也只有读取的权限了。

（3）控制台应用限制更严。

例如常用的 cmd.exe，也就是 MS-DOS 窗口。它使用默认的 ACL，权限如下。

- Administrator：完全控制。
- System：完全控制。
- 交互方式：读取和执行。
- 服务：读取和执行。

（4）匿名账户不再属于 Everyone 组。而 Everyone 组只包含验证后的用户和 Guest

账户。

（5）事件日志的安全增强。

对于应用和定制日志，交互用户只能在本地读取和写入。管理员才能远程访问。

对于系统日志，交互用户只能在本地读取。本地系统、本地服务和网络服务只能在本地写入。只有管理员才能远程读取。

2．增加的有效权限表

具体的操作过程是：打开资源管理器，选择相应的文件或文件夹，然后右击，在弹出的快捷菜单中选择“共享与安全”命令，然后单击“安全”按钮，出现相关用户的权限，然后单击“高级”按钮，就会看到在 Windows Server 2003 中增加的有效权限表。如图 4-8 所示。

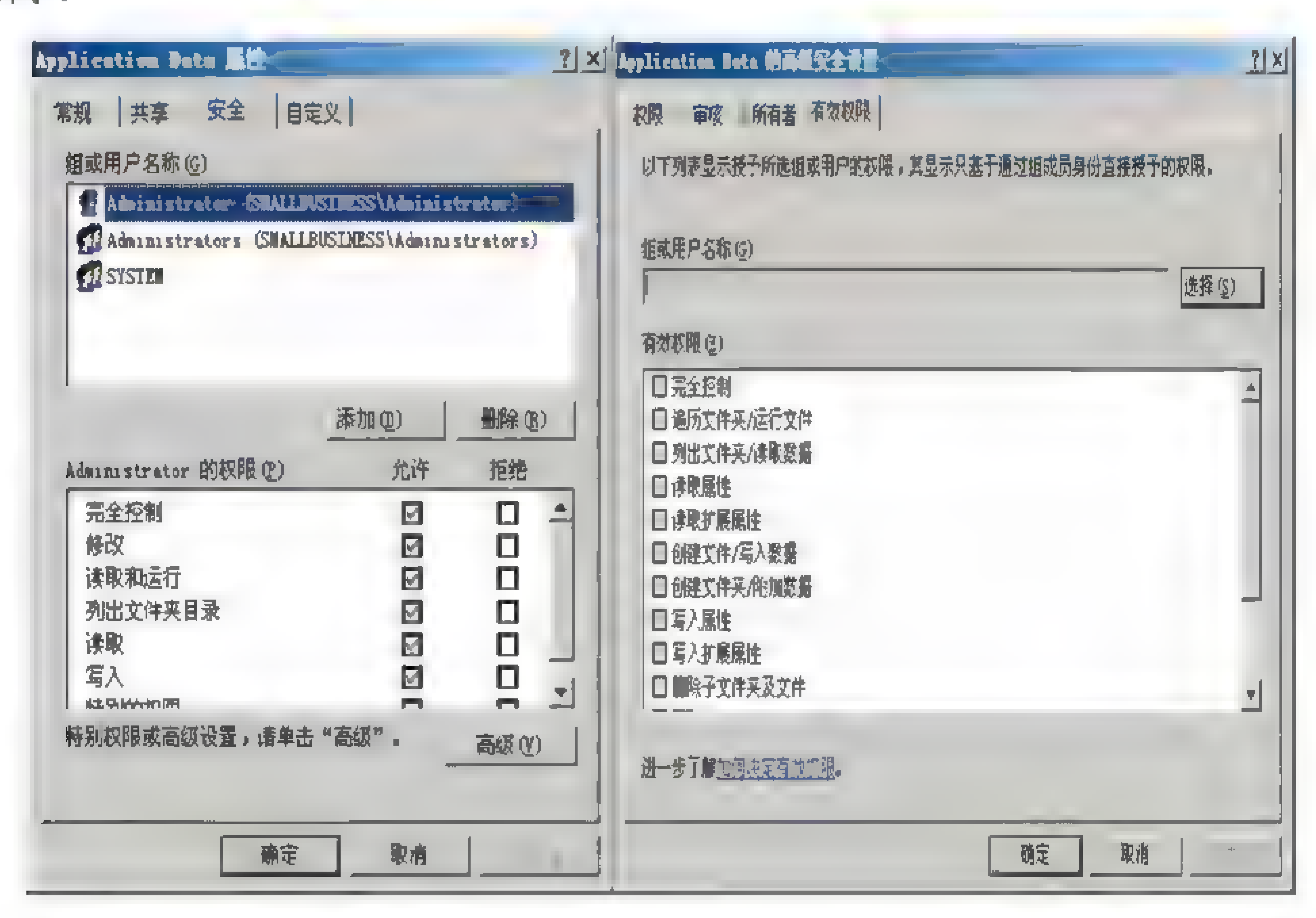


图 4-8

3．默认 AD 安全描述符的改变

对于域对象，如用户、组等，在 Windows Server 2003 中默认安全描述符有了改变。默认的安全描述符可以通过 AD 对象类的属性进行设定。为了使用这个管理单元，用户必须先注册 schmmgmt.dll。具体的操作步骤如下。

（1）选择“开始”→“运行”命令，在“运行”对话框中的“打开”下拉列表框中输入 regsvr32 schmmgmt.dll，如图 4-9 所示。然后单击“确定”按钮，将会弹出注册成功的对话框，如图 4-10 所示。单击“确定”按钮。

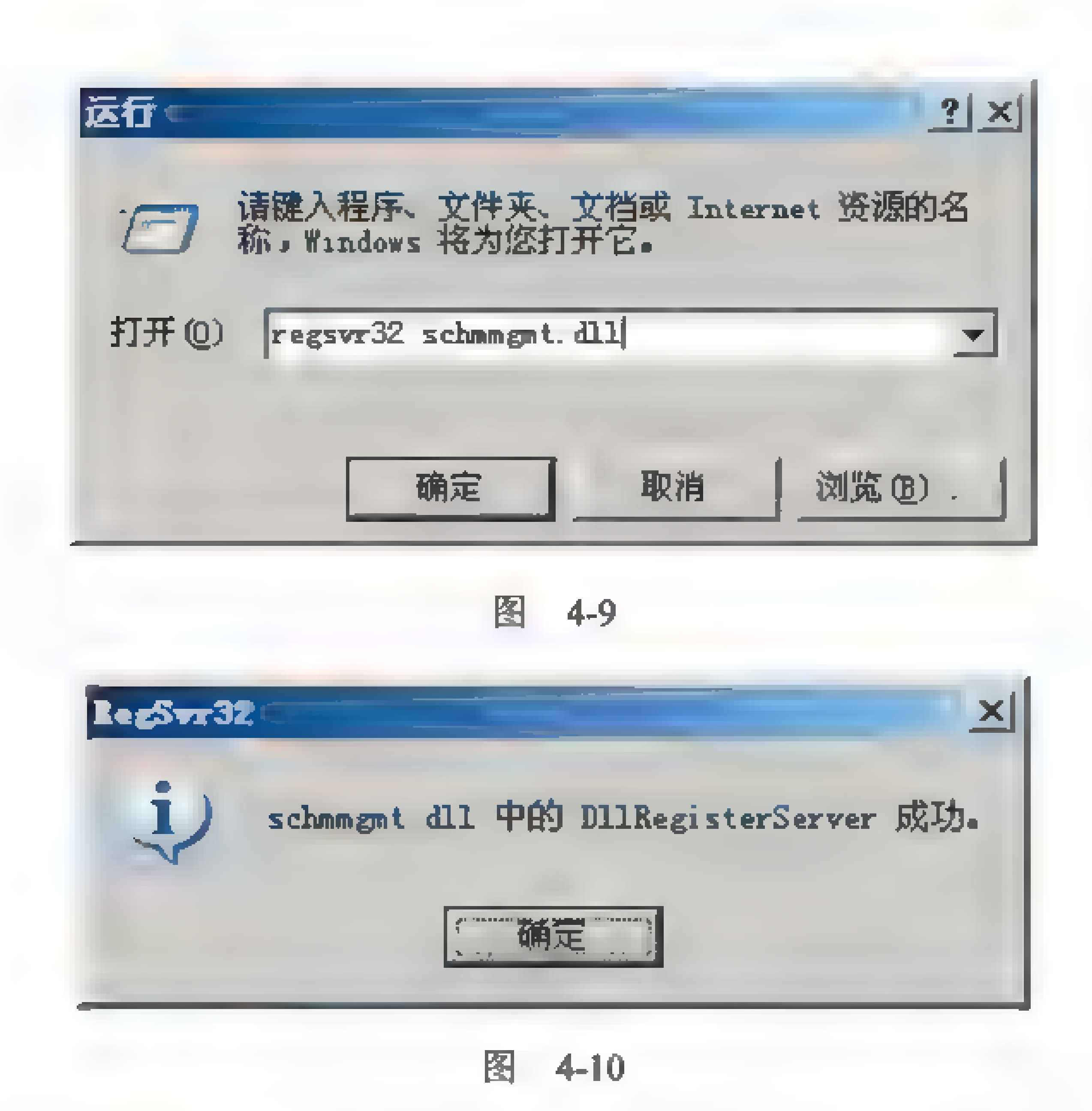


图　4-9

图　4-10

（2）建立活动目录架构 MMC 管理单元，选择“开始”→“运行”命令，在“运行”对话框中的“打开”下拉列表框中输入 mmc，启动控制台，分别如图 4-11 和图 4-12 所示。

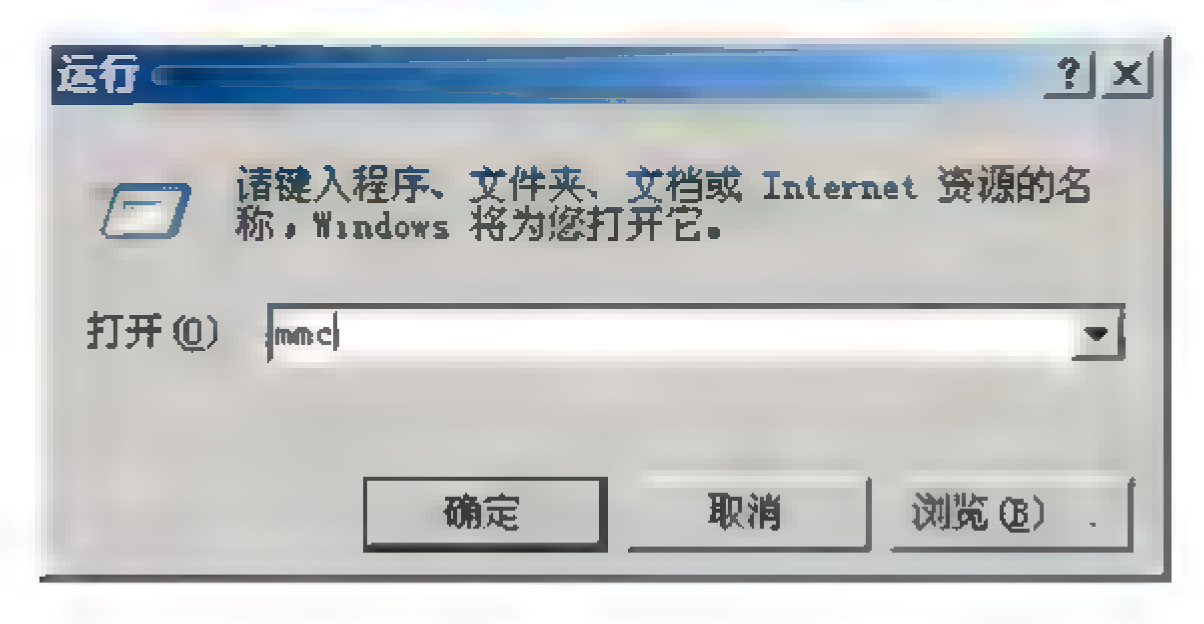


图　4-11

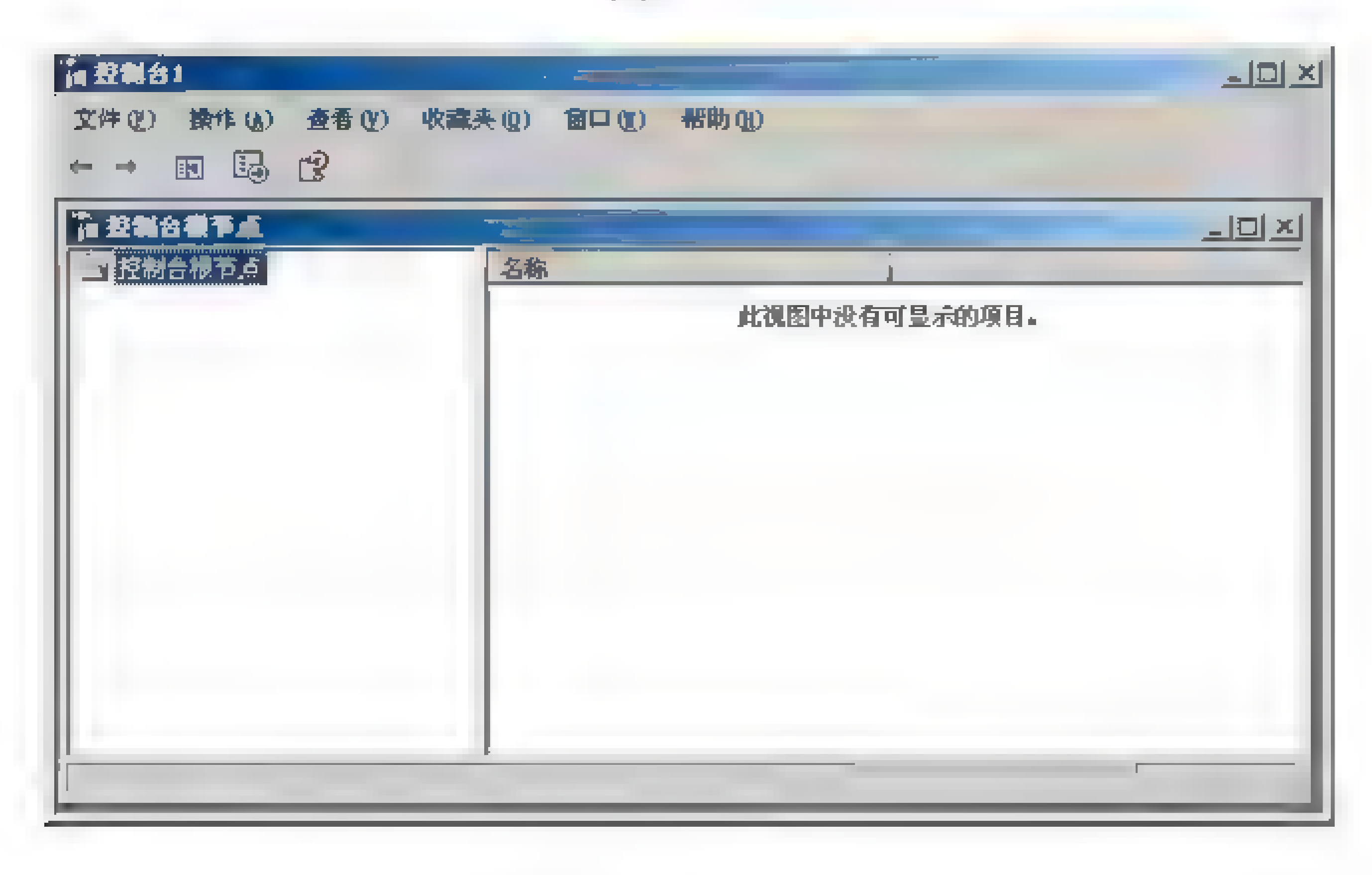


图　4-12

（3）单击控制台中的“文件”菜单，选择“添加/删除管理单元”命令，在打开的对话框中单击“添加”按钮，在列表框中选择 Active Directory 架构选项。如图 4-13 所示，然后单击“确定”按钮。

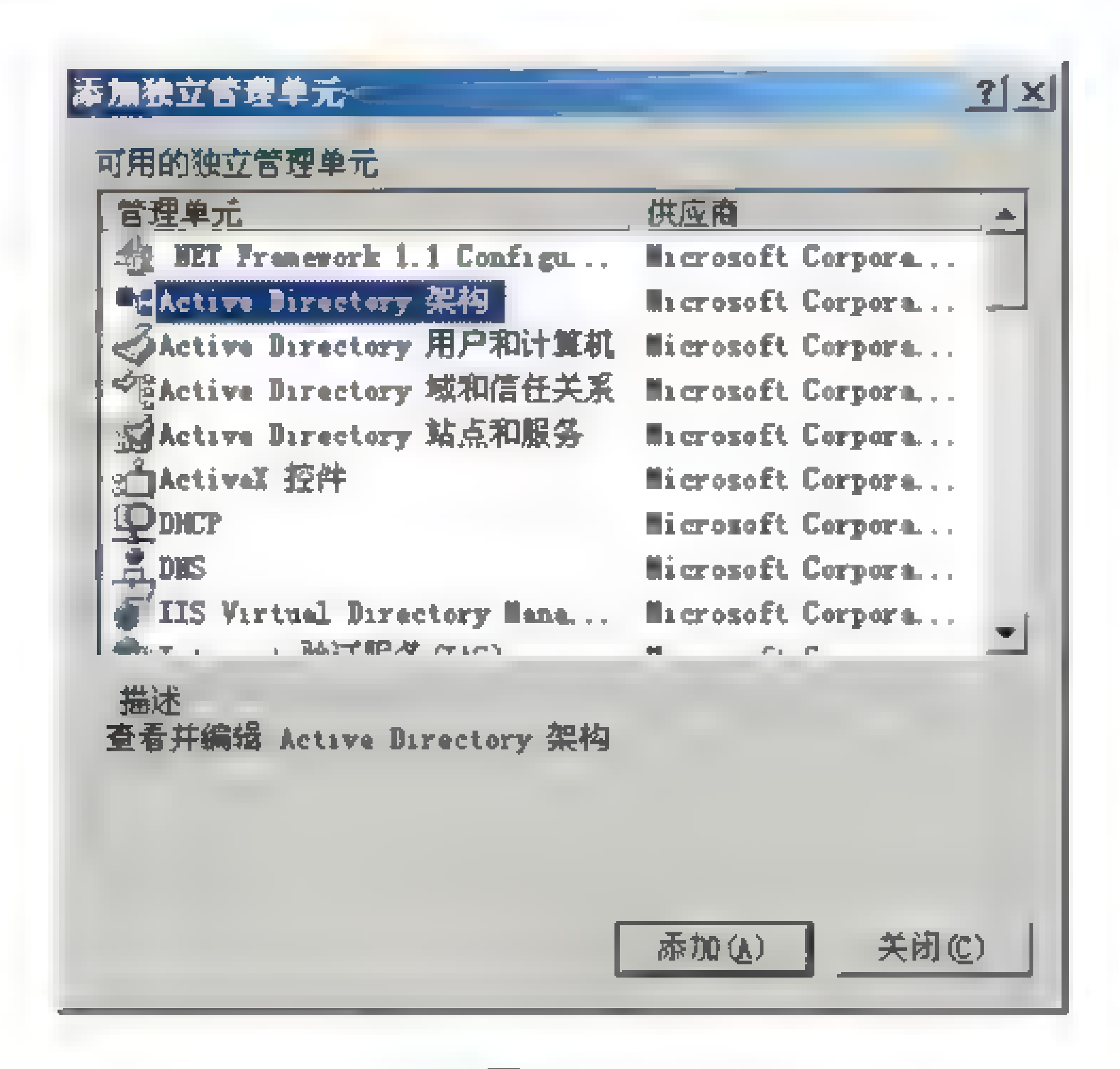


图 4-13

（4）展开类别，为相应的类更改默认的安全性。如图 4-14 所示。



图 4-14

4．AD 对象的配额

AD 对象的配额有助于域控制器抵御拒绝服务攻击，在 2000 版本中是没有 AD 对象配额的。AD 对象配额指定每个独立 AD 的命名上下文和分区。只有 Windows Server 2003 才能执行配额。AD 对象配额使用在 AD 配置分区，如图 4-15 所示。

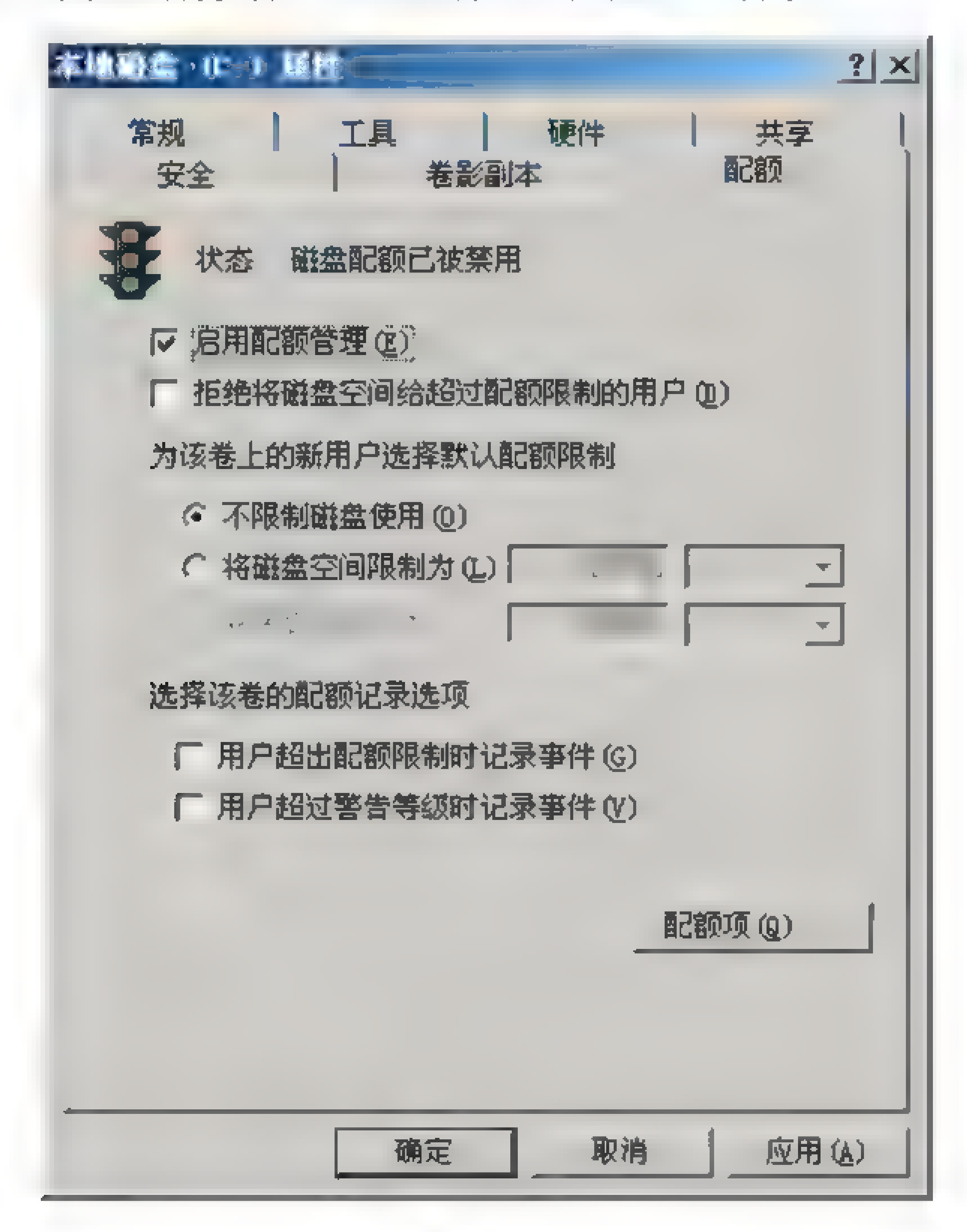


图　4-15

Windows 的磁盘配额是根据所有者属性计算的，授予其他人所有者权限的功能简化了磁盘配额的管理。例如，管理员应用户的要求创建了新的文件（例如复制一些文件，或安装新的软件），使得管理员成为新文件的所有者，即新文件占用的磁盘空间不计入用户的磁盘配额限制。以前，要解决这个问题必须经过烦琐的配置修改，或者必须使用第三方工具。现在，Windows Server 2003 直接在用户界面中提供了设置所有者的功能，这类有关磁盘配额的问题可以方便地解决了（对于使用 NTFS 文件系统的任何类型的操作系统都有效，包括 Windows NT 4.0、2000 和 XP Pro，只要修改是在 Windows Server 2003 上进行就可以）。

另外，在 Windows Server 2003 中还有许多有用的授权工具，分别如下。

- Cacls：用于浏览和更新文件系统的命令行工具。
- Whoami：此命令加/a 参数可以分析用户访问令牌的内容。
- Showpriv：显示用户和组的授权权利的命令行工具。

- Ntrights：能授予或撤销用户和组权利的命令行工具。
- Permcopy：从共享处复制共享权限和 ACL 文件。
- Showacls：列举文件、文件夹和林的访问权限的命令行工具。
- Subinacl：在用户与用户间、全局组与组之间、域与域之间传输安全信息的命令行工具。
- Showmbrs：显示具体组成员的用户名的命令行工具。
- Dsacls：管理 AD 对象的 ACL 的命令行工具。
- Sdcheck：显示 AD 对象的安全描述符的工具。

4.5 Windows Server 2003 的授权管理器

在以前的 Windows 版本中，提供了以对象为中心的授权模型。资源管理器（RM）管理自身的对象集，并使用安全描述符来保护这些对象。每当某个客户端进入并请求访问受 RM 保护的资源时，RM 将模拟该客户端并调用 AccessCheck API。AccessCheck API 将依次查看该客户端的安全令牌、需要访问的对象及该对象的安全描述符。AccessCheck API 向 RM 返回 yes 或 no，由 RM 确定是否允许客户端访问该对象。

在 Windows Server 2003 中引入了一项新的功能，就是基于角色的访问控制（Role Based Access Control，RBAC）架构。这种架构并没有替代原来 Windows 系列平台中的自主访问控制架构，而是将新的关键组件加入到原有的访问控制架构中。图 4-16 给出了 Windows Server 2003 的 RBAC 架构，由授权管理器来完成。授权管理器是 Windows Server 2003 的新功能。

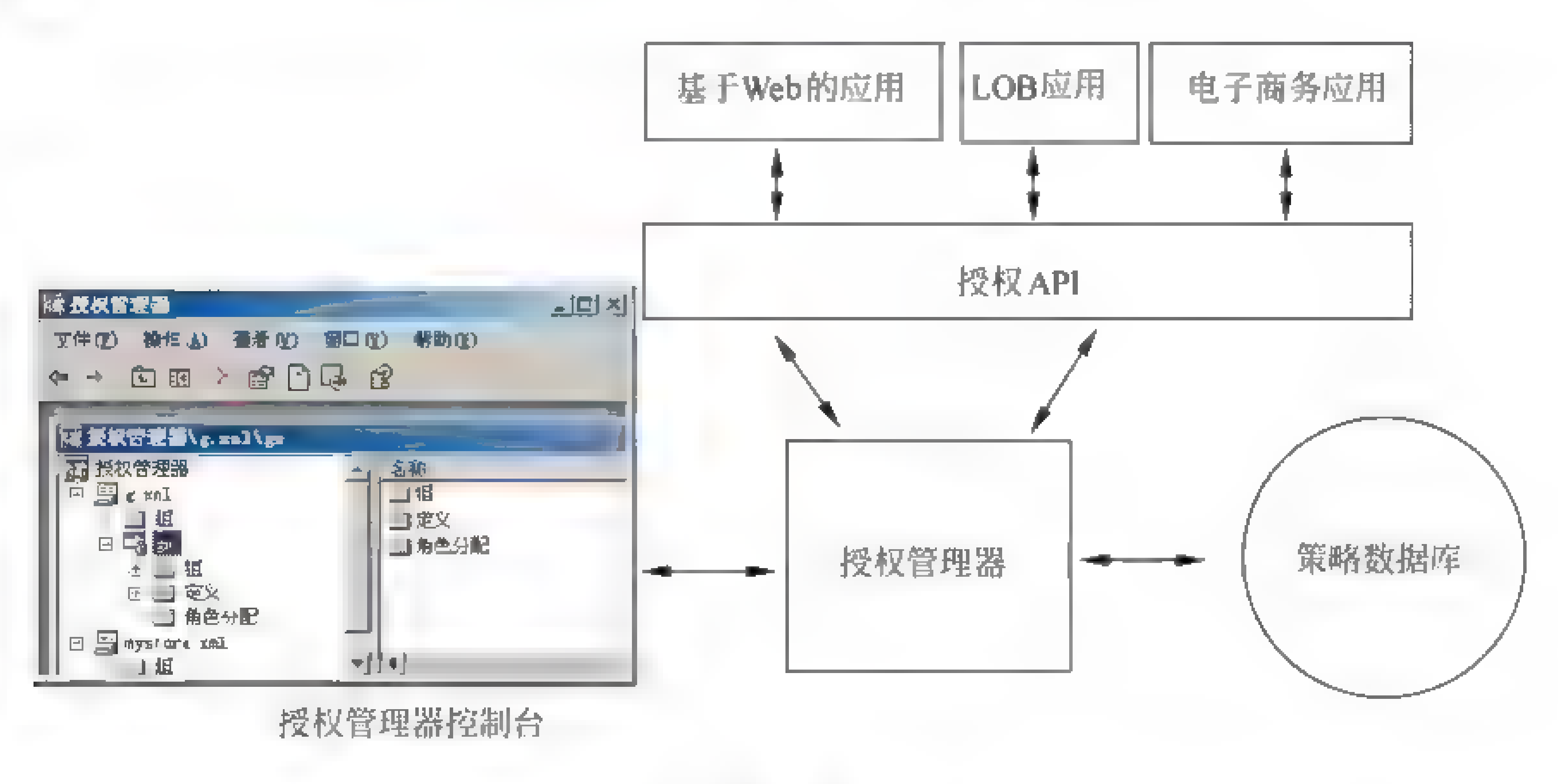


图 4-16

在这个架构中，授权管理器处于核心位置，也就是基于角色的访问控制是由授权管理器完成的。授权管理器（AzMan）是 Windows 的一种通用的、基于角色的新安全体系结构。AzMan 与 COM+ 无关，因此它可以用在任何需要基于角色的授权的应用程序中，包括 ASP.NET Web 应用程序或 Web 服务、基于 .NET Remoting 的客户服务器系统等。在撰写本文时，授权管理器仅在 Windows Server 2003、Windows 2000 的 Service Pack 4 中

提供。

AzMan 有两个部分：运行库和管理 UI。运行库由 AZROLES.DLL 提供，它公开了一组供那些利用基于角色安全的应用程序使用的 COM 接口。管理 UI 是一个 MMC 管理单元，可以通过运行 AZMAN.MSC 或者通过向用户选择的 MMC 控制台中添加授权管理器管理单元对其进行试验，如图 4-17 所示。

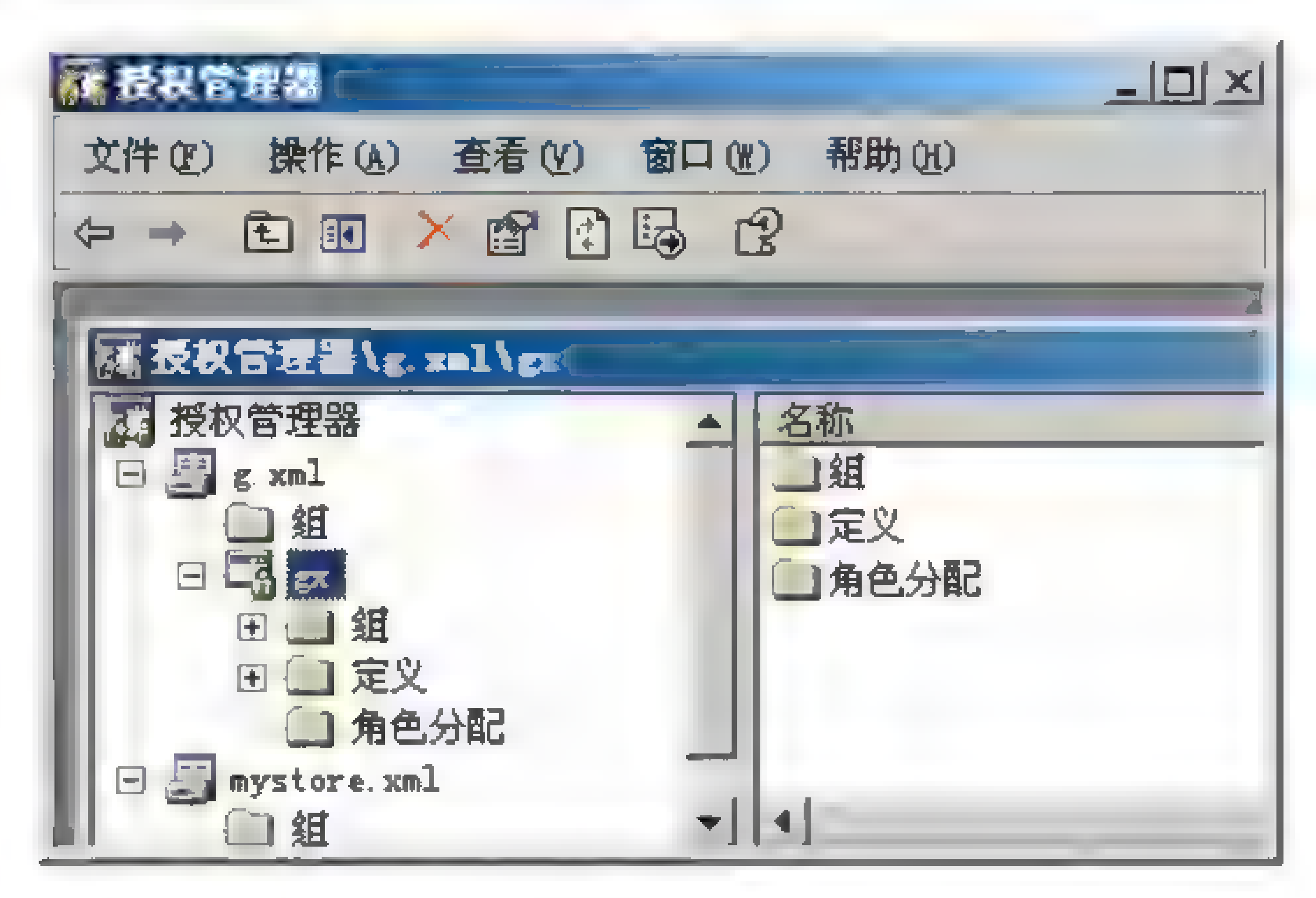


图　4-17

授权管理器为应用程序开发人员提供了一个灵活框架，可将基于角色的访问控制集成到应用程序中。AzMan 管理单元在两种模式下操作：开发人员和管理员。在管理员模式下，没有选择创建存储区或应用程序的自由，并且不能改动应用程序代码所依赖的低级别操作定义。坦白地说，没有什么东西能够妨碍系统管理员进入开发人员模式并完成这些操作，但要点是在管理员模式下，UI 中选项的数量将减少，以简化管理员的工作并帮助他们避免错误。“选项”对话框如图 4-18 所示。

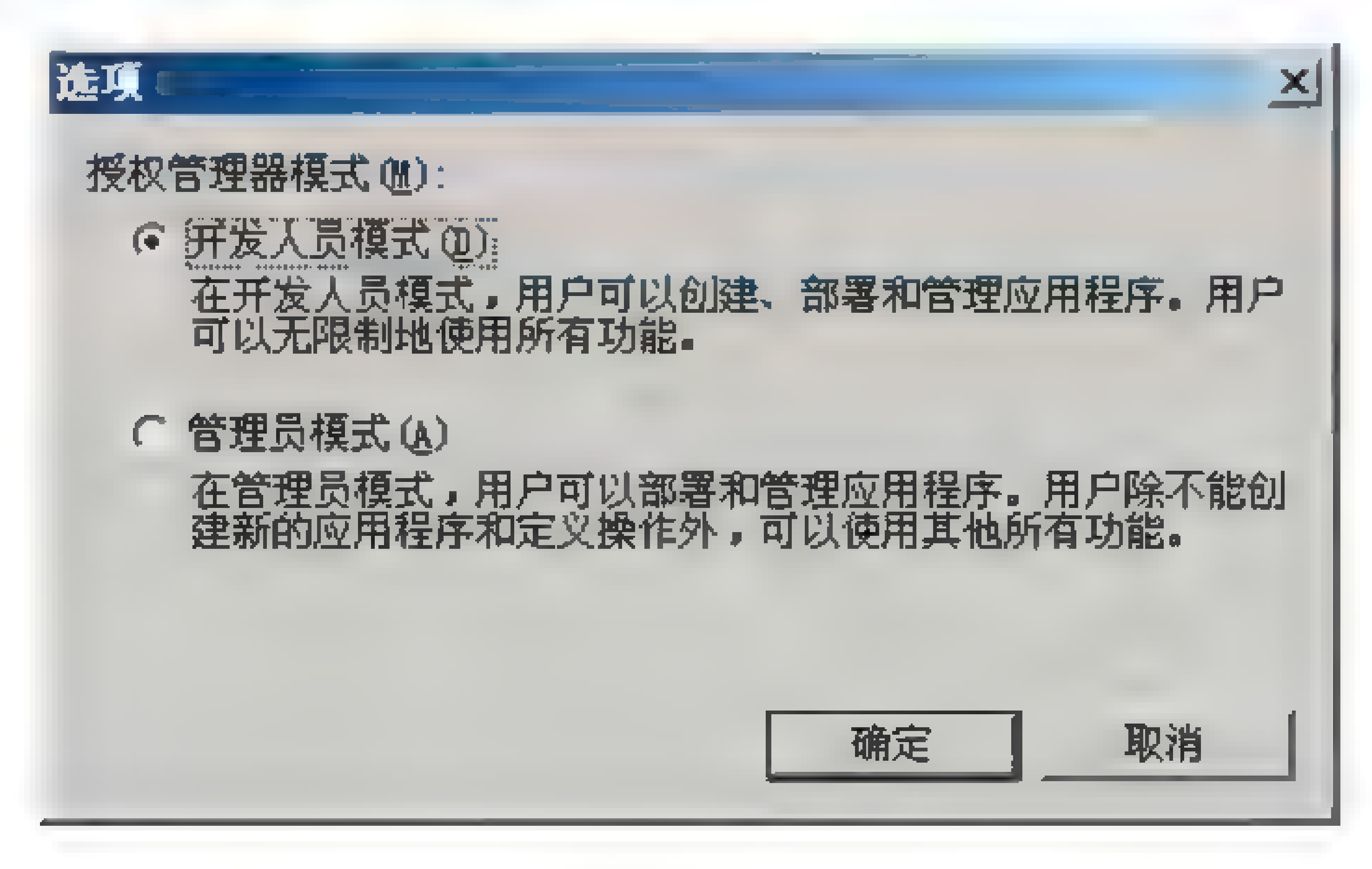


图　4-18

当使用授权管理器的应用程序初始化之后，它将从存储区中加载授权策略信息。在 Windows Server 2003 中，授权管理器允许将授权策略存储在 Active Directory 中，或者以 .xml 格式保存在文件中。包含授权策略存储区的系统的管理员对该存储区具有很高的访问权限，因此授权策略存储区必须存放在可靠的系统中。

当使用 Active Directory 存储区时，授权管理器将为存储区本身创建 Active Directory 对象，并为每个应用程序组、应用程序、操作、任务、角色和范围创建子对象。范围对象可以包含在该范围内创建的任务、角色和组。

另外，授权管理器还允许授权策略以. xml 格式保存在 NTFS 文件系统（该系统受 ACL 保护）的文件中。XML 存储区可位于用作授权管理器服务器的计算机上，也可以保存在远程位置。为了支持重命名对象，.xml 格式包含了全局唯一标识符（GUID），这就导致不需要也不能直接编辑 .xml 文件。可以通过授权管理器 MMC UI 来编辑存储区，也可以通过可供诸如 VBScript 和 JScript 之类的脚本语言使用的授权管理器接口来编辑存储区。

在安装时，调用相应的 API 创建授权存储区、创建操作和任务，可能还要创建应用程序需要的某些初始角色。在运行时，应用程序将初始化授权管理器并将其连接到授权存储区，然后与该应用程序特有的存储区部分建立连接。

Windows Server 2003 中的基于角色的访问控制为业务流程类型的应用程序提供了简化的开发模型。管理员和开发人员都可以受益于这个合理的框架，有效地为组织结构和业务流程建模。

4.6 Windows Server 2003 的安全模式

在现在的网络应用中，信息的安全则显得更加重要。下面从用户的角度对 Windows 的安全性做一些探讨。

Windows Server 2003 通过一系列的管理工具，以及对用户账号、口令的管理，对文件、数据授权访问，执行动作的限制，以及对事件的审核达到 C2 级安全。从用户的角度看，通过这一套完整、可行、易用而并不烦琐的措施可以达到较好的效果。

Windows 的安全机制的基础是所有的资源和操作都受到选择访问控制的保护，许多安全机制不是外加的，而是建立在操作系统内部的，可以通过一定的设置使文件和其他资源免受在存放的计算机上工作的用户和通过网络接触资源的用户的威胁（如破坏、非法的编辑等）。安全机制甚至提供基本的系统功能，例如设置系统时钟。对用户账号、用户权限及资源权限的合理组合，可以有效地保证安全性。

4.6.1 Windows Server 2003 的安全策略

作为自身考虑的安全策略，对于用户而言，Windows 有以下几种管理手段，这些对安全性有着极大的影响。

- 用户账号和用户密码。
- 域名管理。

- 用户组权限。
- 共享资源的权限。

1. 用户账号和用户密码

Windows 的安全机制通过请求分配用户账号和用户密码来帮助保护计算机及其资源。给值得信任的使用者，按其使用的要求和网络所能给予的服务分配合适的用户账号，并且给其容易记住的账号密码。使用对账号的用户权力的限制及对文件的访问管理权限的策略，可以达到对服务器的数据的保护。其中用户账号有用户名、全名和描述三个部分。用户名是用户账号的标识，全名是对应用户名的全称，描述是对用户所拥有的权限的较具体的说明。组包括组名和描述两个部分，组名是标识，描述是说明。一定的用户账号对应一定的权限，Windows Server 2003 对权限的划分更细，如备份、远程管理和更改系统时间等，通过对用户的授权（在规则菜单中）可以细化一个用户或组的权限。用户的账号和密码有一定的规则，包括账号长度、密码的有效期、登录失败的锁定及登录的历史记录等，通过对这些的综合修改可以保证用户账号的安全使用。

2. 域名管理

以 Windows Server 2003 组建的网络是一个局域网范围的网。所谓“域”是指网络服务器和其他计算机的逻辑分组，凡是在共享域范围内的用户都使用公共的安全机制和用户账号信息。每个用户有一个账号，每次登录的是整个域，而不是某一个服务器。即使在物理上相隔较远，但在逻辑上可以在一个域上，这样便于管理。在网络环境下，使用域的管理就显得更为有效。这里应该注意到，在 Windows Server 2003 中，关于域的所用的安全机制信息或用户账号信息都存放在目录数据库（称为安全账号管理器，SAM）中。目录数据库存放在服务器中，并且复制到备份服务器中。通过有规律的同步处理，可以保证数据库的安全性、有效性。在用户每次登录时，通过目录数据库检查用户的账号和密码。所以在对 Windows Server 2003 进行维护时应该特别小心目录数据库的完整性，一般来讲只有管理员才具有对此的编辑权限。

域的最大优点是域中的控制器服务器形成了共享的安全机制和用户账号信息的单个管理单元，大大地节省了管理员及用户的精力和时间，在管理上较方便，也显得集中。在使用“域”的划分时，应该注意到“域”是建立在一个子网范围内，其基础是相互之间的信任度。由 Server 2003 组网区别于一般的 TCP/IP 的组网，TCP/IP 是一种较松散的组网形式，靠路由器完成子网之间的寻径通信；而 Windows Server 2003 组网是一种紧密的联合，服务器之间是靠安全信任建立它们的联系的。主从关系、委托关系是建立在信任度上的。

3. 用户组权限

管理员一般根据用户访问网络的类型和等级给用户分组。组包括“全局组”和“本地组”。全局组由一个域的几个用户账号组成，所谓全局是指可以授予该组使用多个（全局）域资源的权力和权限。全局组只能在域中创建。本地组有用户账号和一个或多个域中的全局组构成，这些用户在同一个账号下，只有非本地域的用户和全局组处于受托域中时，才可以将其添加到本地组中。本地组可以包含用户和全局组，但不能包含其他本地组。

Windows Server 2003 域控制器包含内置本地组，它决定了用户登录到域控制器时可以进行的操作，域控制器上的内置本地组给管理员在管理域安全机制方面增加了隐患。因为每一个内置本地组都有一套预先确定的权限，这些权限自动地应用于添加到该组中的每个用户账号，假若需要对组中的某些用户的权限做一些修改，可以在用户管理器中进行。建议在使用其默认权力时，对用户进行仔细的筛选，防止在组员中有信用度不高的用户，而对网络资源造成损坏。

4. 共享资源权限

Windows Server 2003 允许用指定他人共享的资源。资源共享后，可以通过网络限制某些用户对它的访问权限，这称为共享权限的限制。针对不同的用户，可以利用资源共享及资源权限来创建不同的资源安全级别。Windows Server 2003 的较大特点在其文件系统。在 NTFS 文件系统中，可以使用权限对单个文件进行保护，并且可以把该权限应用到本地访问和网络访问中。在 NTFS 卷上，可以对文件设置文件权限，对目录设置目录权限。用于指定可以访问的组和用户及允许的访问等级时，NTFS 卷的共享权限与文件及目录的权限共同起作用。共享目录时，通过共享目录设置的权限允许用户连接到共享资源，反之改变设备可以中断与用户的共享资源的连接。资源所有者或管理员使用 NTFS 的共享目录的默认权限（完全控制），可以使用目录与文件权限来管理文件的安全性问题。

4.6.2 在网络中 Windows Server 2003 的安全性

作为一种针对网络应用的操作系统，安全性通过其本身的内部机制得到了一定的基本保证，例如用户账号、用户密码、共享资源权限和用户管理器等。在实践中的应用也证明了 Windows Server 2003 的安全性的可靠程度还能支持应用。但是，随着网络的不断扩大，以及通过 Internet 互联，资源的共享和系统的安全、用户的隐私、运行的效率这些矛盾日益的突出。在用户获得较大的自由度和灵活性而与世界各地的人和计算机通信的同时，在另一方面增加了系统的冒险性——有人也可以自由通过网络访问你的机器或资源，甚至于你的隐私，给你的系统增加负担，降低运行效率。为了更好地发挥服务器的作用，同时减轻其在安全性上的冒险性，有必要在 Server 2003 安全模型的基础上结合 IIS 的安全机制，这样在客观上讲，可以较好地解决上述的矛盾。

在网络中，有三种方式可以访问 Windows Server 2003 服务器。

1. 通过用户账号、密码和用户组方式登录到服务器

在服务器允许的权限内对资源进行访问、操作。这种方式的可控制性较强，可以针对不同的用户。

Windows Server 2003 系统首先必须在 Server 2003 中拥有一个账号，其次规定该账号在系统中的权力和权限。

在 Windows Server 2003 系统中，权力专指用户对整个系统能够做的事情，如关掉系统、往系统中添加设备及更改系统时间等。权限专指用户对系统资源所能做的事情，如对某文件的读、写控制，对打印机队列的管理。Server 2003 系统中有一个安全账号数据库，其中

存放的内容有用户账号及该账号所具有的权力等。用户对系统资源所具有的权限则与特定的资源一起存放。

用户登录过程

在没有用户登录时，可以看到屏幕上显示一个对话框，提示用户登录在 Windows NT 系统中。实际上，Windows Server 2003 系统中有一个登录进程。当用户在开始登录时，按下 Ctrl+Alt+Del 组合键，Windows Server 2003 系统启动登录进程，弹出登录对话框，让用户输入账号名及口令。按下 Ctrl+Alt+Del 组合键时，Windows Server 2003 系统保证弹出的登录对话框是系统本身的，而不是一个貌似登录对话框的应用程序，以防止被非法窃取用户名及口令。

所以，在登录时，无论屏幕上是否有登录对话框，一定要按下 Ctrl+Alt+Del 组合键，以确保弹出的对话框是 Windows Server 2003 系统的登录对话框，此过程就是强制性登录过程。登录进程收到用户输入的账号和口令后，就查找安全账户数据库中的信息。如果账户及口令无效，则用户的登录企图被拒绝；如果账户及口令有效，则把安全账户数据库中有关该账户的信息收集在一起，形成一个存取标识。

存取标识中的主要内容有：

- 用户名及 SID。
- 用户所属的组及组 SID。
- 用户对系统所具有的权力。

然后 Windows Server 2003 就启动一个用户进程，将该存取标识与之连在一起，这个存取标识就成了用户进程在 Windows Server 2003 系统中的通行证。

用户无论做什么事情，Windows Server 2003 中负责安全的进程都会检查其存取标识，以确定其操作是否合法。

用户成功地登录之后，只要用户没有注销自己，其在系统中的权力就以存取标识为准，Windows Server 2003 安全系统在此期间不再检查安全账户数据库。这主要是考虑到效率。

存取标识的作用相当于缓存，只不过存取标识缓存的是用户安全信息，使得系统不必再从硬盘上查找。安全账户数据库是由域用户管理器来维护的，在某个用户登录后，有可能管理员会修改其账户及权力等，但这些修改只有在用户下次登录时才有效，因为 Windows Server 2003 安全系统在用户登录后只检查存取标识，而不是检查安全账户数据库。比如 User1 已登录到了 Windows Server 2003 系统中，管理员发现其缺少了某种权力，就用域用户管理器做了相应的修改，那么，除非 User1 重新登录一次，否则 User1 仍无法享有该权力。

存取标识包含的内容并没有访问权限，而存取标识又是用户在系统中的通行证，那么 NT 如何根据存取标识控制用户对资源的访问呢？

原来，给资源分配的权限作为该资源的一个属性，与资源一起存放。比如有目录为 D:\Files，对其指定 User1 只读，User2 可完全控制，则这两个权限都作为 D:\Files 目录的属性与该目录连在一起。在 Server 2003 内部以访问控制列表的形式存放。ACL 中包含了每个权限的分配，以访问控制项来表示。ACE 中包含了用户名及该用户的权限。比如上面提到的这个例子中，D:\Files 的 ACL 中有两个 ACE，分别是“User1:只读”和“User2:完全控制”。当 User1 访问该目录时，Server 2003 安全系统检查用户的存取标识，与目录

的 ACL 对照，发现用户存取标识中的用户名与 ACL 中有对应关系且所要求的权限合法，则访问获得允许，否则，访问被拒绝。

2. 在局部范围内通过资源共享的形式登录网络

这种方式建立在 NETBIOS 的基础之上。对共享的访问不能经过路由器，范围被限制在一个子网范围内，在使用的灵活性上受到限制，通过对共享资源的共享权限的控制达到安全保护。但不能针对不同的用户，当一个用户在通过共享对某一个资源进行操作时（这时共享权限有所扩大），其他用户乘虚而入，而造成对资源的破坏。例如设置 NTFS 的安全性：

Windows Server 2003 获得美国政府的 C2 安全性认证在很大程度上取决于 NTFS 文件系统具有很强的安全性。

NTFS 文件系统的安全性体现在：除非用户拥有必要的权限，否则不能够访问 NTFS 上的文件。

NTFS 将所有的文件和目录都看成是对象，并为它们设置权限，可以在每一个文件级和用户级层次上进行访问控制。NTFS 还具有审计能力，可以跟踪哪些文件曾被成功地访问，哪些文件曾被恶意访问。

为 NTFS 设置安全性包括为 NTFS 上的文件或目录设置权限。

选中要设置权限的文件或者目录右击，在弹出的快捷菜单中选择“属性”命令，打开如图 4-19 所示的对话框，选择“安全”选项卡。在对话框中，可以添加组和用户，然后可以设置相关的权限，也可以设置特别权限。

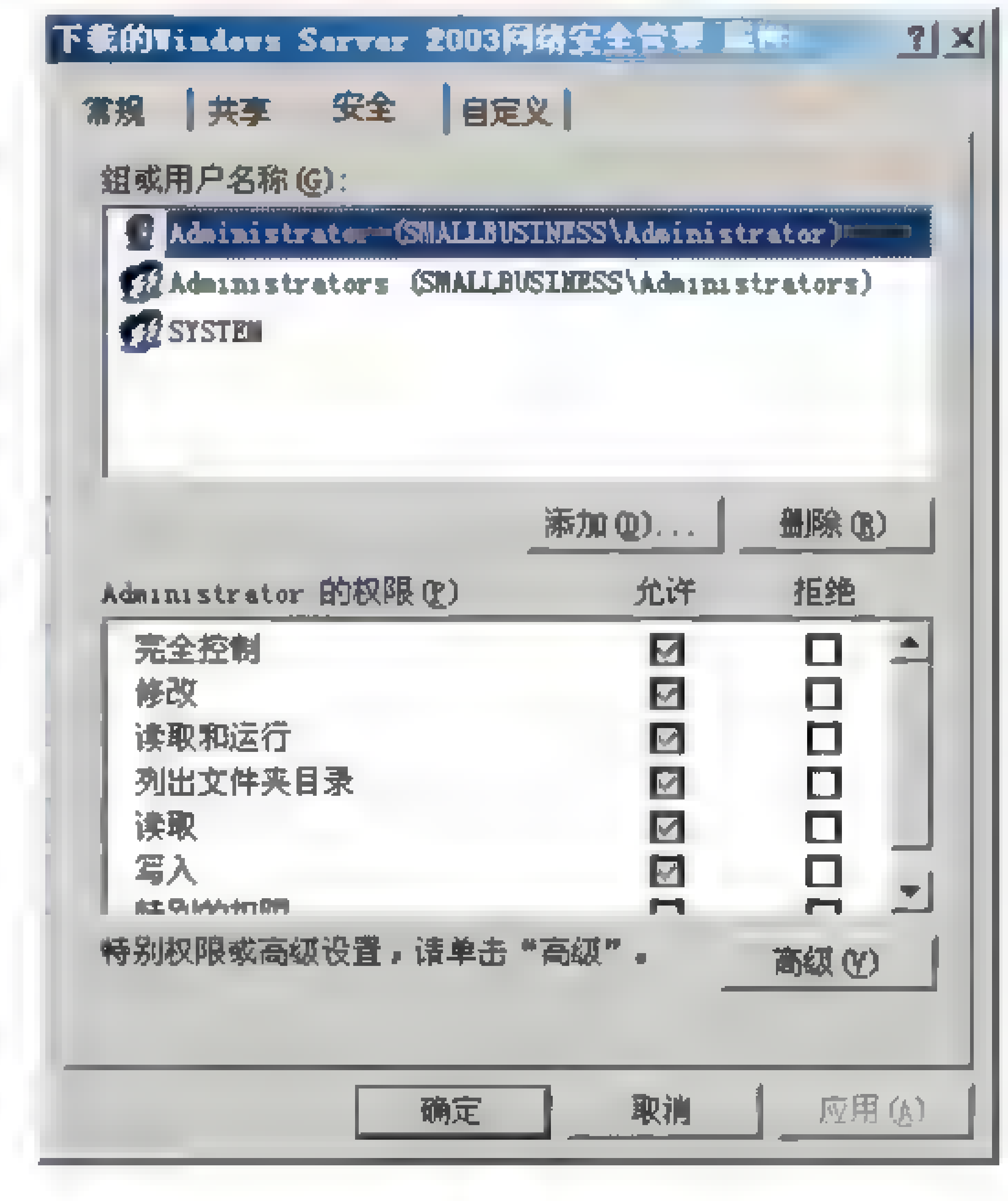


图 4-19

3. 在网络中通过 TCP/IP 协议对服务器进行访问

目前典型应用有 FTP、HTTP 和 WWW 等。通过对文件权限的限制和对 IP 的选择，对登录用户的认证可以在安全性上做到一定的保护。但由于 Windows 是微软的产品，其透明度并不高，安全隐患有可能就隐藏于此。

（1）Windows Server 2003 本身有可能存在 Bug，一旦被发现，就有可能造成损失。

（2）由于网络的日益庞大，使通过 INTERNET 访问某个国家的机密成为可能，假如在编写网络操作系统的同时，为以后通过 TCP/IP 入侵留下隐藏的人为的漏洞。

下面针对上述三种情况在安全措施上作一些介绍。

1）设置用户账号

Windows Server 2003 的用户管理器指定允许某些用户或用户组可以在服务器上操作，可以控制对 Web 节点的访问。可以通过 Web 客户请求提供在完成请求之前定制的 IIS 用户名和密码，这样可以进一步控制在网络中对服务器的访问。通过公共的管理工具中的“策略”设置用户权力，在“用户管理器”中配置在计算机上授权用户所进行的操作。用户 Basic 身份验证时，用户所要求用于 Internet 服务的权力。若使用 Windows Server 2003 Challege/Response 身份验证，用户使用 Internet 服务则需要有“从网络访问本机”的权力。

2）设置必要的 W W W 目录访问权限

在 Internet Server manager 中创建 Web 发布目录（文件夹）时，可以为定义的主目录或虚拟目录及其中所有的文件夹设置访问权限，这些权限是 NTFS 文件系统提供的权限之外的部分，其中的权限是只读，只执行，这样可以防止用户修改。

3）通过 IP 地址控制访问权限

可以配置 IIS 以允许或拒绝特定的 IP 地址访问你的服务器或整个网络。在实际应用中，对于一些未知的用户，若从安全的角度出发，可以通过 IP 设置排出。假若在日志分析中通过分析可以发现某些用户或用户组有不良倾向或侵犯倾向，那 Administrator 可以通过 IIS 设置，不再允许这些用户或组的 IP 地址访问本机及本网络。

4）使用 SSL 保护数据在网络中的传输

在网络应用中，数据在网络上传输的安全是关系到整个网络应有安全的重要问题。使用密码技术保护数据从服务器到客户端的双向传输，从某种意义上说也是保护 SERVER 2003 资源不受侵犯的有效途径。SSL 为 TCP/IP 连接提供数据加密，服务器身份验证和消息的完整性。它被视为 Internet 上的 Web 浏览器和服务器的标准安全措施。SSL 加密的传输较之未加密的传输速度要慢，为了避免整个节点的性能受到影响，所以一般考虑对于较敏感的信息数据才采取 SSL 加密。目前使用较多的是身份验证、信用卡和电子银行等业务。

4.7　Windows Server 2003 的安全管理

最严格的安全防护也不能防止所有的安全事故。操作系统提供的安全功能中很重要的一点就是责任，它确保任何实体的动作将唯一用该实体标识。一个实体可以是一个人，一个操作系统资源，或者一个外部系统，例如计算机或者网络。

操作系统的责任服务把所有的安全相关事件同一个标识联系起来。根据以下两个指标评估责任：

- 机制用来分配责任的强度。
- 操作系统基于这个信息进行决策的能力。

在 Windows Server 2003 中，安全管理是一项非常重要的服务，它新增加了软件安全策略这一新功能。并且如前所述，在安全架构上增加了许多安全的组件来确保操作系统的安全。但为了保证服务器的安全，管理员还需要根据企业或单位的实际情况来配置相关的安全策略。在 Windows Server 2003 中，安全管理主要体现在三个主要方面：安全策略管理、安全补丁管理和相关的审核管理。相应的安全设置如图 4-20 所示。

图 4-20 中的大部分策略都可以使用组策略来完成。还有一些配置需要使用安全配置编辑器和一些配置工具，分别用三种不同的图形来表示。

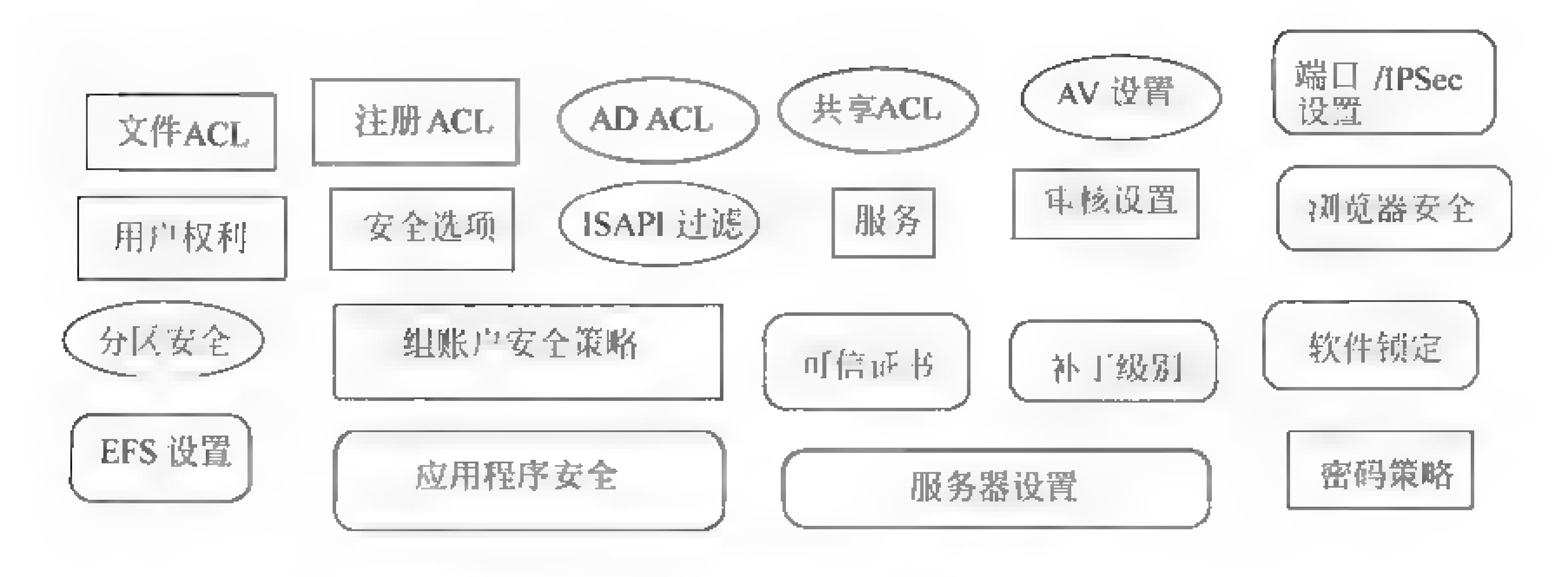


图 4-20

4.7.1 Windows Server 2003 组策略

管理员可以使用组策略来定义用户工作的环境。用户和计算机设置都在组策略中，Windows Server 2003 加强了组策略设置，增加了更新设置和软件安全策略。可以给站点、域或组织单元容器来配置组策略。组策略可以为站点集中设置策略，对于域和组织单元可以对不同的计算机和用户账户配置不同的策略。启动组策略编辑器，如图 4-21 所示。

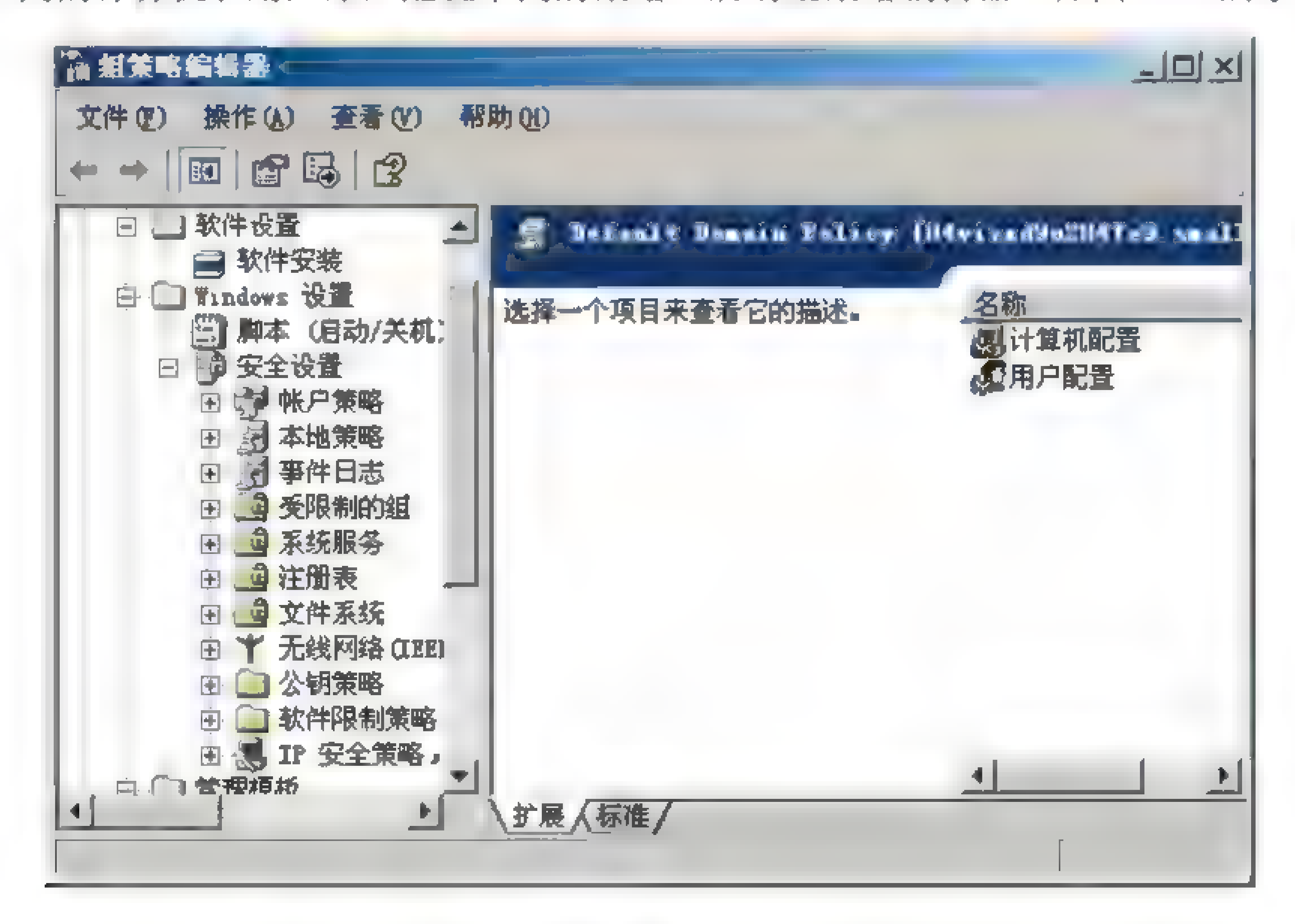


图 4-21

在组策略中有许多策略，在这里只介绍安全设置。可以对本地计算机、域和网络进行安全设置。主要包括用户账户策略和审核策略及用户权利和事件日志。

首先使用 Windows Server 2003 中的安全模板，选择“开始”→“运行”命令，在“运行”对话框中的“打开”下拉列表框中输入 mmc.exe。出现控制台，然后单击“文件”按

钮，选择“添加/删除管理单元”，出现如图 4-22 所示对话框，选择安全模板。然后单击“关闭”按钮，会出现安全管理模板的控制台。如图 4-23 所示。



图 4-22

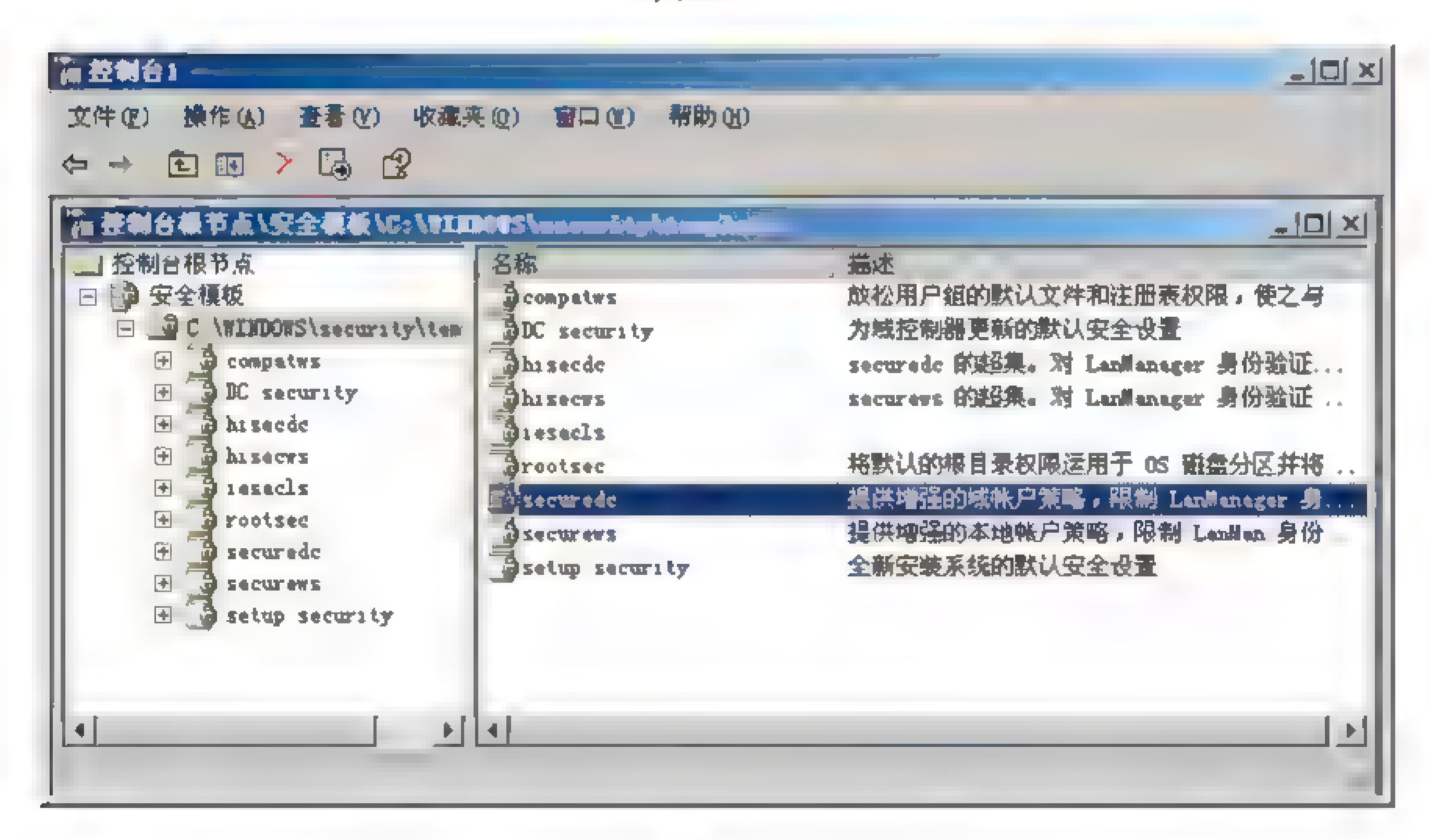


图 4-23

从图 4-23 可知，可以通过安全模板，对用户、域控制器和域控制器管理单元设置相关的策略。

用户账户的保护一般主要围绕着密码的保护来进行。为了避免用户身份由于密码被破解而被夺取或盗用，通常可采取诸如提高密码的破解难度、启用账户锁定策略、限制用户登录、限制外部连接及防范网络嗅探等措施。在 Windows Server 2003 系统的账户和本地策略中包括账户策略和本地策略两个方面，而其中的“账户策略”又包括密码策略、账户锁

定策略和 Kerberos 策略三个方面。因为前面已经详细地介绍了 Kerberos 策略，这里只介绍前两种。另外的“本地策略”包括审核策略、用户权限分配和安全选项三部分。下面分别予以介绍。

1．账户策略

（1）密码策略的设置

提高密码的破解难度主要是通过采用提高密码复杂性、增大密码长度及提高更换频率等措施来实现，但这常常是用户很难做到的，对于网络中的一些安全敏感用户就必须采取一些相关的措施，以强制改变不安全的密码使用习惯。

在 Windows 系统中可以通过一系列的安全设置，并同时制定相应的安全策略来实现。在 Windows Server 2003 系统中，可以通过在安全策略中设定“密码策略”来进行。Windows Server 2003 系统的安全策略可以根据网络的情况，针对不同的场合和范围进行有针对性的设定。例如可以针对本地计算机、域及相应的组织单元来进行设定，这将取决于该策略要影响的范围。以域安全策略为例，其作用范围是企业网中所指定域的所有成员。在域管理工具中运行“域安全策略”工具，然后就可以针对密码策略进行相应的设定。

密码策略也可以在指定的计算机上用本地安全策略来设定，同时也可在网络中特定的组织单元通过组策略进行设定。

密码策略作用于域账户或本地账户，其中就包含以下几个方面：

- 强制密码历史。
- 密码最长使用期限。
- 密码最短使用期限。
- 密码长度最小值。
- 密码必须符合复杂性要求。
- 用可还原的加密来存储密码。

以上各项的配置方法均需根据当前用户账户类型来选择。默认情况下，成员计算机的配置与其域控制器的配置相同。在图 4-24 中需要定义相关密码策略，在相应策略处右击，选择属性，就可以设置这些策略了。

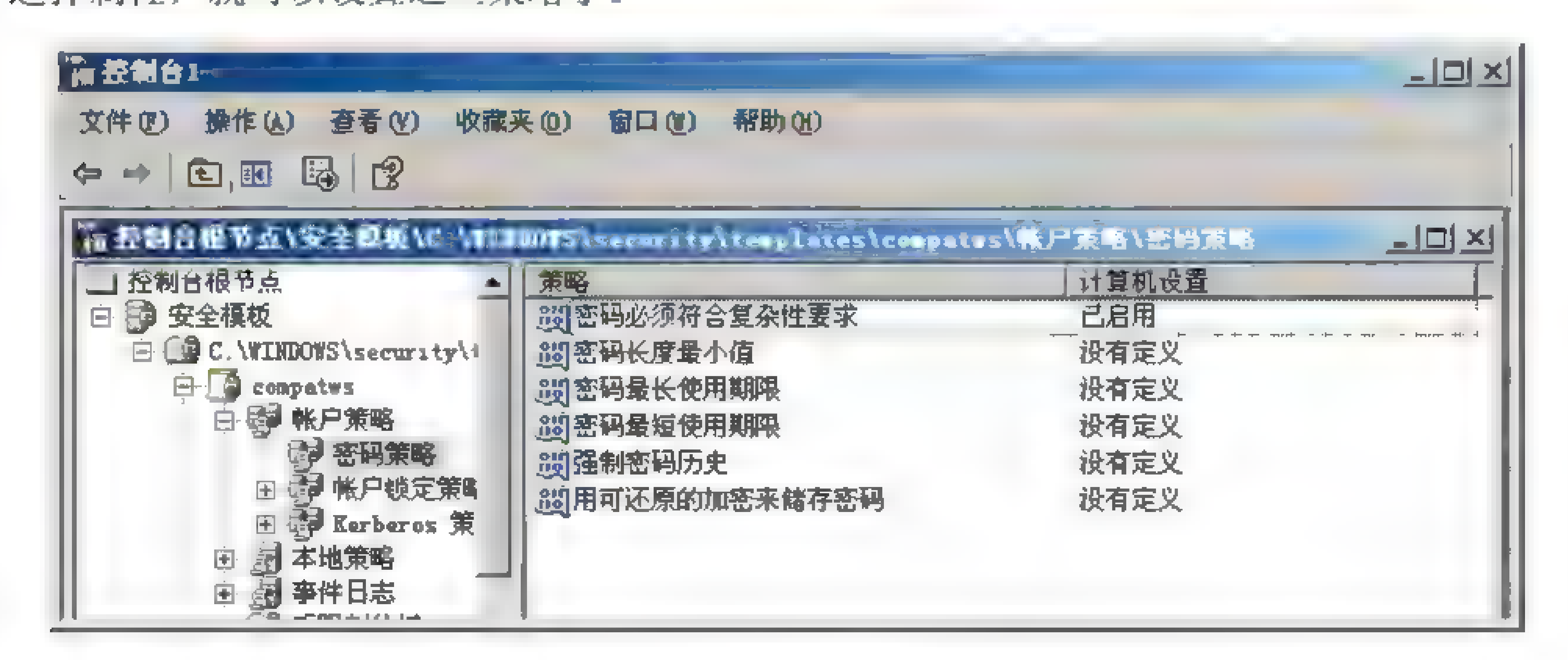


图　4-24

（2）账户锁定策略

账户锁定是指在某些情况下（例如账户受到采用密码词典或暴力猜解方式的在线自动登录攻击），为保护该账户的安全而将此账户进行锁定。使其在一定的时间内不能再次使用，从而挫败连续的猜解尝试。

Windows 2003 系统在默认情况下，为方便用户起见，这种锁定策略并没有进行设定，此时，对黑客的攻击没有任何限制。只要有耐心，通过自动登录工具和密码猜解字典进行攻击，甚至可以进行暴力模式的攻击，那么破解密码只是一个时间上的问题。账户锁定策略设定的第一步就是指定账户锁定的阈值，即锁定前该账户无效登录的次数。一般来说，由于操作失误造成的登录失败的次数是有限的。在这里设置锁定阈值为 3 次，这样只允许 3 次登录尝试。如果 3 次登录全部失败，就会锁定该账户。

但是，一旦该账户被锁定后，即使是合法用户也无法使用了。只有管理员才可以重新启用该账户，这就造成了许多不便。为方便用户起见，可以同时设定锁定的时间和复位计数器的时间，这样在 3 次无效登录后就开始锁定账户，锁定时间为 30 分钟。以上的账户锁定设定，可以有效地避免自动猜解工具的攻击，同时对于手动尝试者的耐心和信心也可造成很大的打击。锁定用户账户常常会造成一些不便，但系统的安全有时更为重要。

用于域账户或本地用户账户，它们确定某个账户被系统锁定的情况和时间长短。主要包含以下三个方面。

- 账户锁定时间：该安全设置确定锁定的账户在自动解锁前保持锁定状态的分钟数。有效范围为 0～99 999 分钟。如果将账户锁定时间设置为 0，那么在管理员明确将其解锁前，该账户将被锁定。如果定义了账户锁定阈值，则账户锁定时间必须大于或等于重置时间。默认值是“无”。因为只有当指定了账户锁定阈值时，该策略设置才有意义。
- 账户锁定阈值：该安全设置确定造成用户账户被锁定的登录失败尝试的次数。无法使用锁定的账户，除非管理员进行了重新设置或该账户的锁定时间已过期。登录尝试失败的范围可设置为 0～999。如果将此值设为 0，则将无法锁定账户。对于使用 Ctrl+Alt+Delete 键或带有密码保护的屏幕保护程序锁定的工作站或成员服务器计算机上，失败的密码尝试计入失败的登录尝试次数中。默认值为 0。
- 复位账户锁定计数器：该安全设置确定在登录尝试失败计数器被复位为 0（即 0 次失败登录尝试）之前，尝试登录失败之后所需的分钟数。有效范围为 1～99 999 分钟。如果定义了账户锁定阈值，则该复位时间必须小于或等于账户锁定时间。默认值为“无”，因为只有当指定了“账户锁定阈值”时，该策略设置才有意义。

配置方法也是通过双击，或单击选择某账户锁定策略选项，然后再单击右键，选择属性选项，打开类似图 4-25 所示对话框，在其中进行配置即可。

2. 审核策略

从安全的观点看，审核是重现安全相关事件以支持对事件的原因和影响的检查。审核跟踪或者系统日志信息可以被用来判断是否有违反政策的事情发生或者是否有值得怀疑的事情。老练的侵入探测产品使用操作系统的审核踪迹作为分析的基础。审核踪迹还提供了跟踪复杂的安全性事故的来源和提供任何补救行动可能需要的证据的能力。

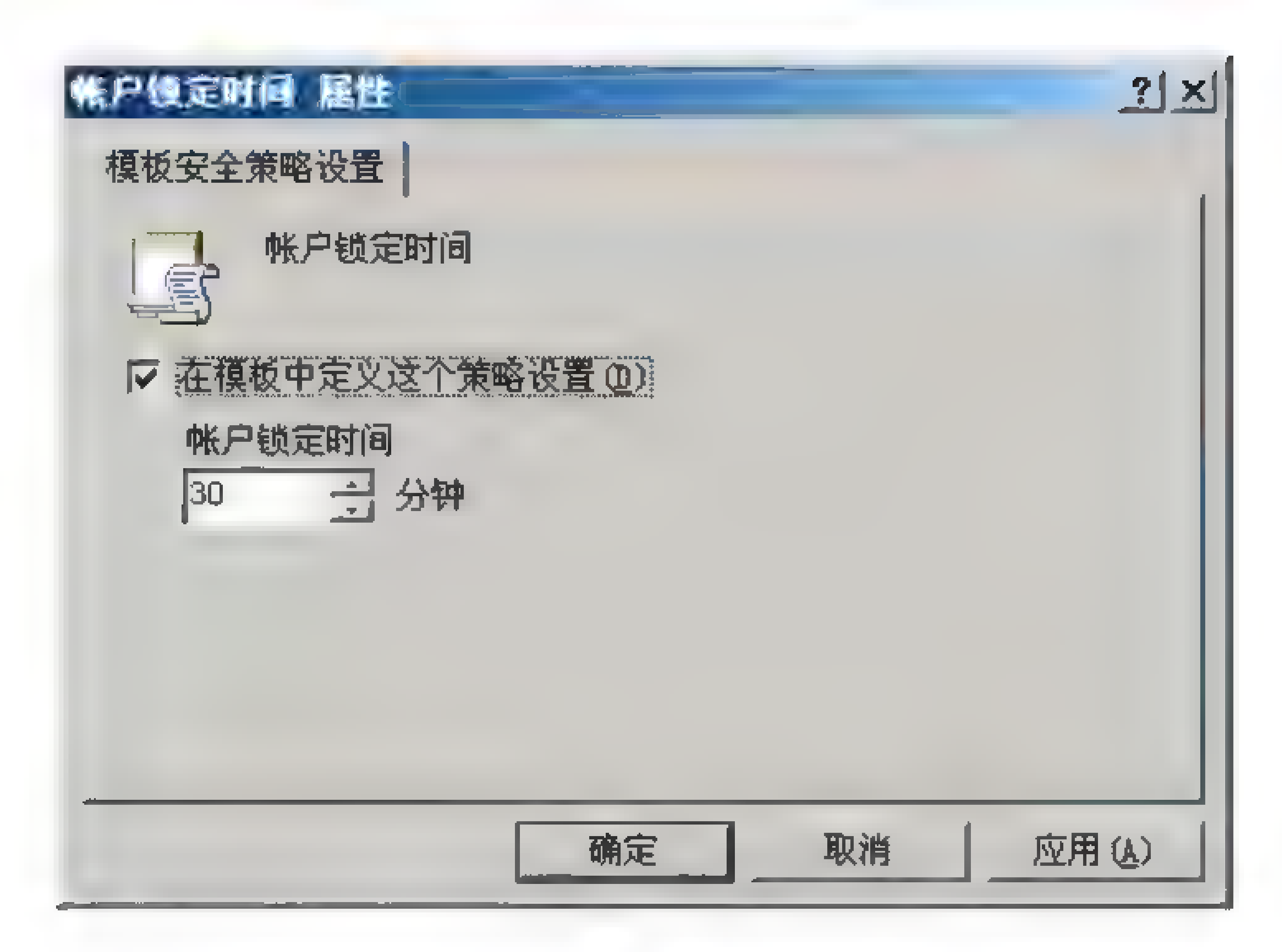


图 4-25

可以根据下面的原则评估操作系统的安全审核能力：

- 支持的安全相关事件的宽度和深度。
- 可以用来保护审核踪迹的机制强度（黑客在闯入系统以后的第一步往往是关闭审核功能或者删除审核日志）。
- 对处理大量由操作系统产生的审核数据的支持。

Windows Server 2003 提供了事件日志（EventLogging）。事件日志可以被配置在系统级别和对象级别，记录安全相关事件。系统事件包括登录和退出登录，文件和对象访问，用户权利的使用，用户和组管理，安全政策改变，重新启动和关机，系统错误，以及进程跟踪。文件和对象审核可以被控制在单个文件，目录，或者如果需要的话，也可以是驱动器。

在管理操作系统时，审核策略是非常重要的，Windows Server 2003 系统审核策略可以应用于本地计算机和域进行配置，如图 4-26 所示。详细信息窗格中要更改审核策略设置的事件类别。然后在对话框中设置即可完成相关事件的审核。

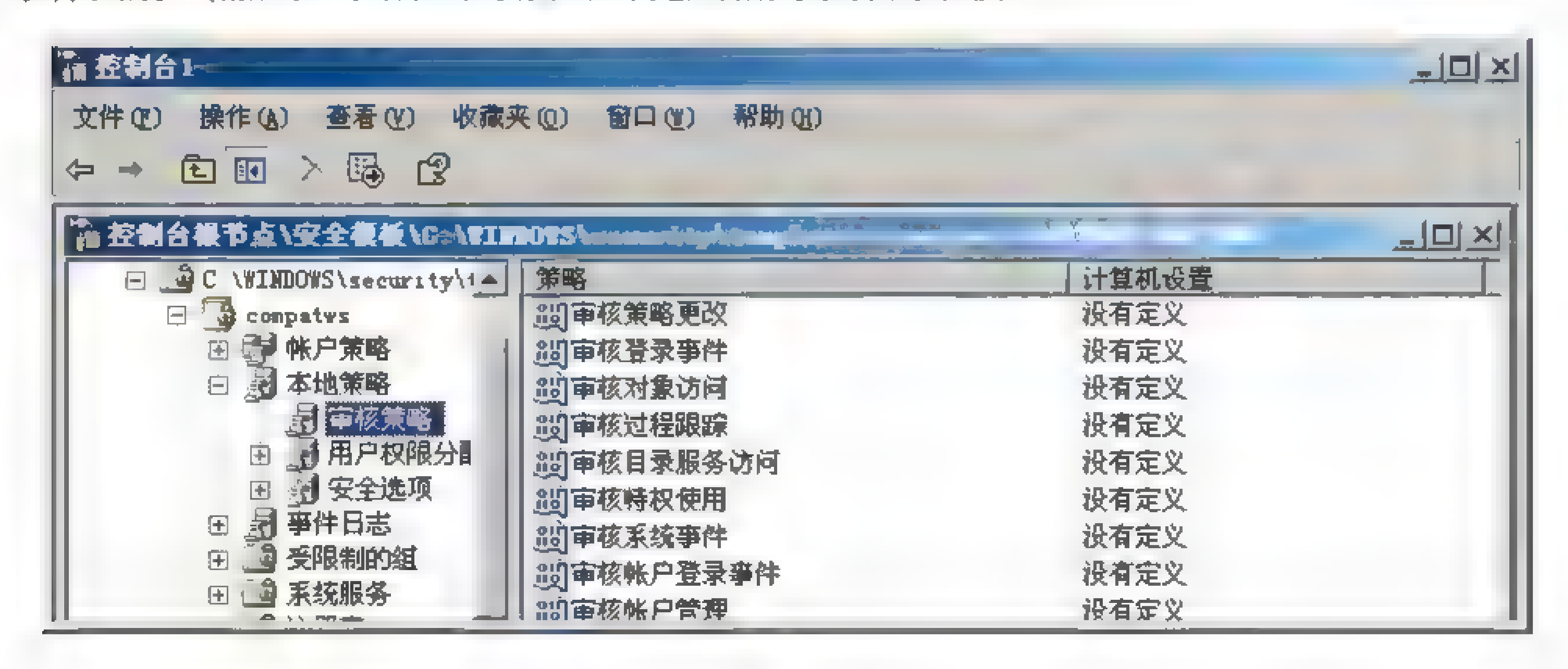


图 4-26

另外，在组策略中，还可以配置用户权限分配，如图 4-26 所示，具体操作也是在详细窗格中双击相应的策略，进行配置即可。

上面所介绍的只是方法，具体要设置哪些行为的权限，就要看用户的具体情况了。虽然这需要有较大的耐心一项一项的配置，但它却是一劳永逸的做法。图 4-27 所示的每一项策略，都需要具体配置，这样才能打造出一道坚不可摧的系统防线。另外，像事件日志等其他的策略配置，配置过程都非常相似。读者可以根据自己的实际需要进行配置。

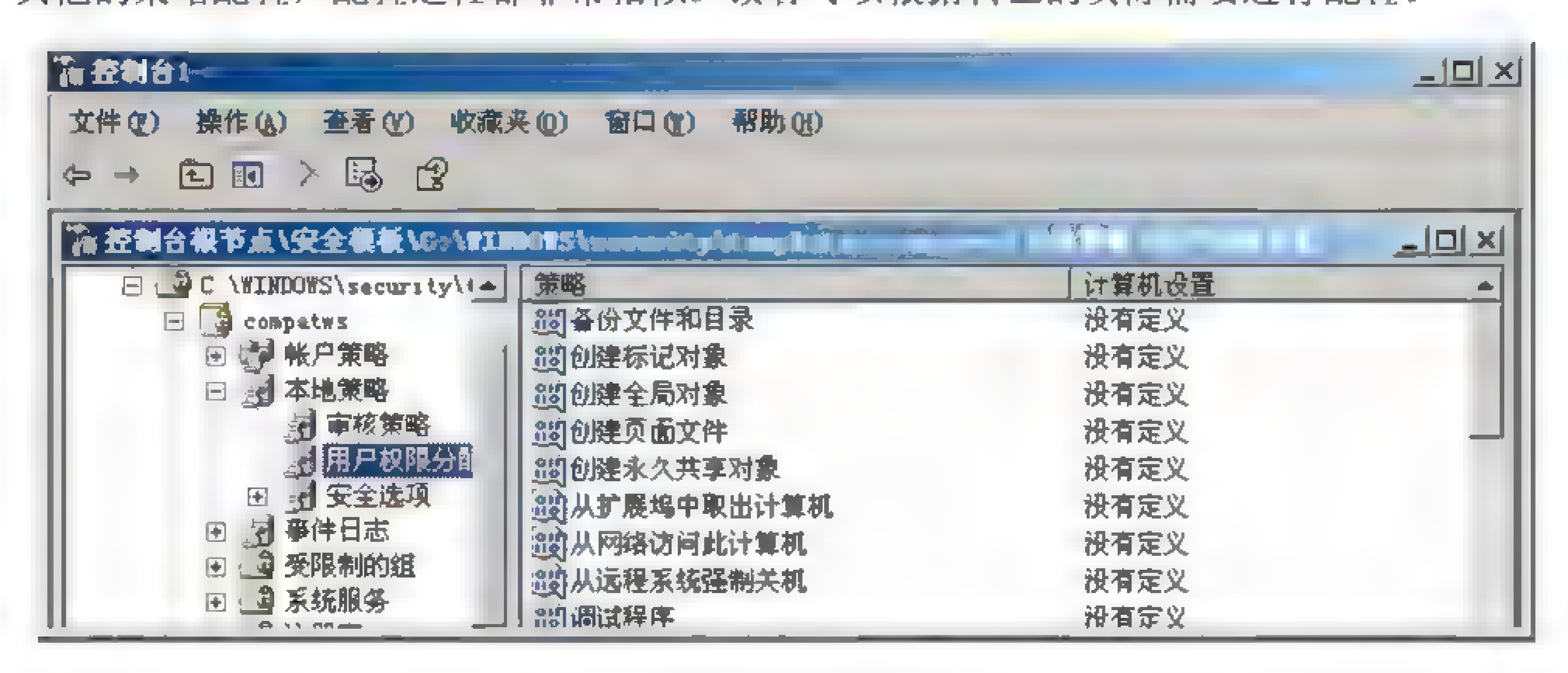


图 4-27

4.7.2 安全分区

从安全性的角度看，每一个系统实体（用户或者计算机资源）被分配一个安全分区，或者叫做域。一个安全域是一个逻辑结构，由实体被授权访问的所有对象组成。一个用户域也许包括存储空间、I/O 设备、应用程序，以及其他元素。一个进程域只包含那些被授权使用的系统资源（如数据、存储空间和 I/O 等）这个用户以及他们可以访问的资源的分区和分段与 Windows Server 2003 操作系统中使用的网络管理域是两个不同的概念。

在系统实体创建的时候就定义一个域，发生在用户被添加到一个计算机系统的时候。该定义基于某些在较大的系统上为不同个体或者用户种类定义安全政策的信任模型上。域分离强制实体实现机制，确保任何实体都真正在它定义的域上操作，同时仍然坚持修改域定义的政策。

强制域分离的操作系统机制的效力基于限制对授权域外访问的机制的强度。对进行中的域分离强制来说，操作特性也是一个关键，如果域很难管理，系统管理员通常默认使用很宽的域定义，这样可以最小化改变的影响，但是同时也大大地降低了能提供的安全级别。集成到目录结构中，都基于角色的域定义的支持和其他高级机制都是很重要的因素。

Windows Server 2003 通过维护进程间的地址分离提供了进程孤立。实体间的所有访问和通信都通过仲裁接口。该仲裁是内核提供的，在一个用户编程不能使用的保护地址空间中操作。所有的内核功能包含在一个单一的模块中，即安全参考模块（Security Reference Module）。这样把所有安全相关的功能集中到一个从一开始就为安全而设计单一模块中。

通过这个模块，Windows Server 2003 内核控制任何应用程序可以访问的资源。例如，用个普通用户的许可运行的应用程序不能访问其他应用程序，或者具有更高权限的用户保留的内存或文件系统资源，内核自动强制实现这个功能。

Windows Server 2003 在分配系统缓冲给 个用户之前会清除该缓冲，并且维护 个文件使用指针以防止磁盘上文件块的重用。这样可以防止用户之间的不可预测的信息流。另外，Windows Server 2003 提供应用程序在返还交换空间给操作系统之前清除它们的能力，在系统关机的情况下也是一样。这些是很重要的存储空间重用功能，可以防止攻击者读取其他用户的应用程序留下的信息。

4.7.3 安全分区加密文件系统

Windows Server 2003 支持利用 EFS 技术对任意文件或文件夹进行加密，这是一种核心的文件加密技术，基于 NTFS 系统，只有 NTFS 系统的磁盘才能使用此技术。加密后的文件或文件夹就不可被除此用户以外的任何用户访问。这样就能更好地保护自己的敏感数据和重要文件。选择要加密的文件，右击，在弹出的快捷菜单中选择“属性”命令，打开如图 4-28 所示对话框。单击“高级”按钮，打开如图 4-29 所示对话框，选中加密内容以保护数据。

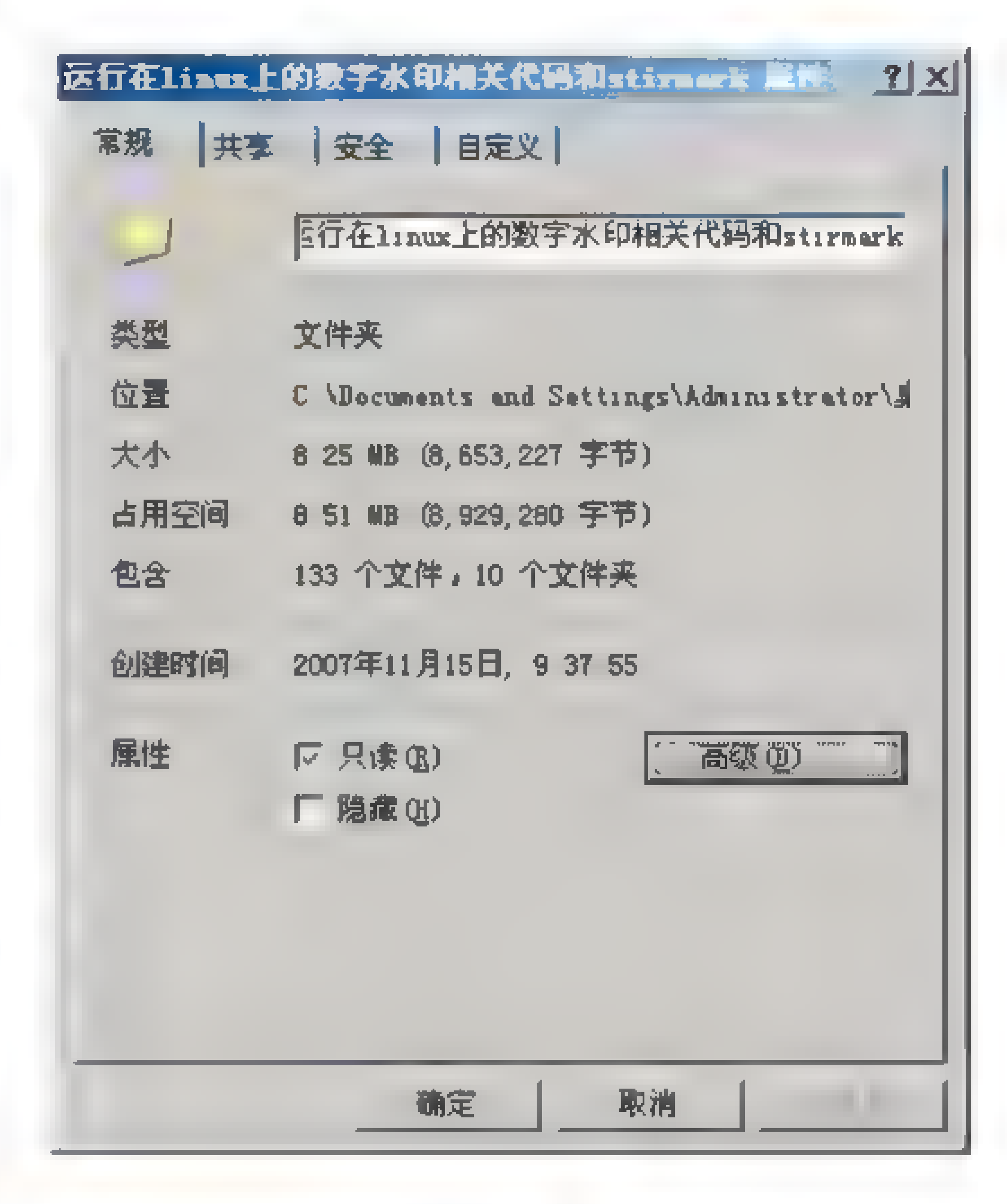


图 4-28

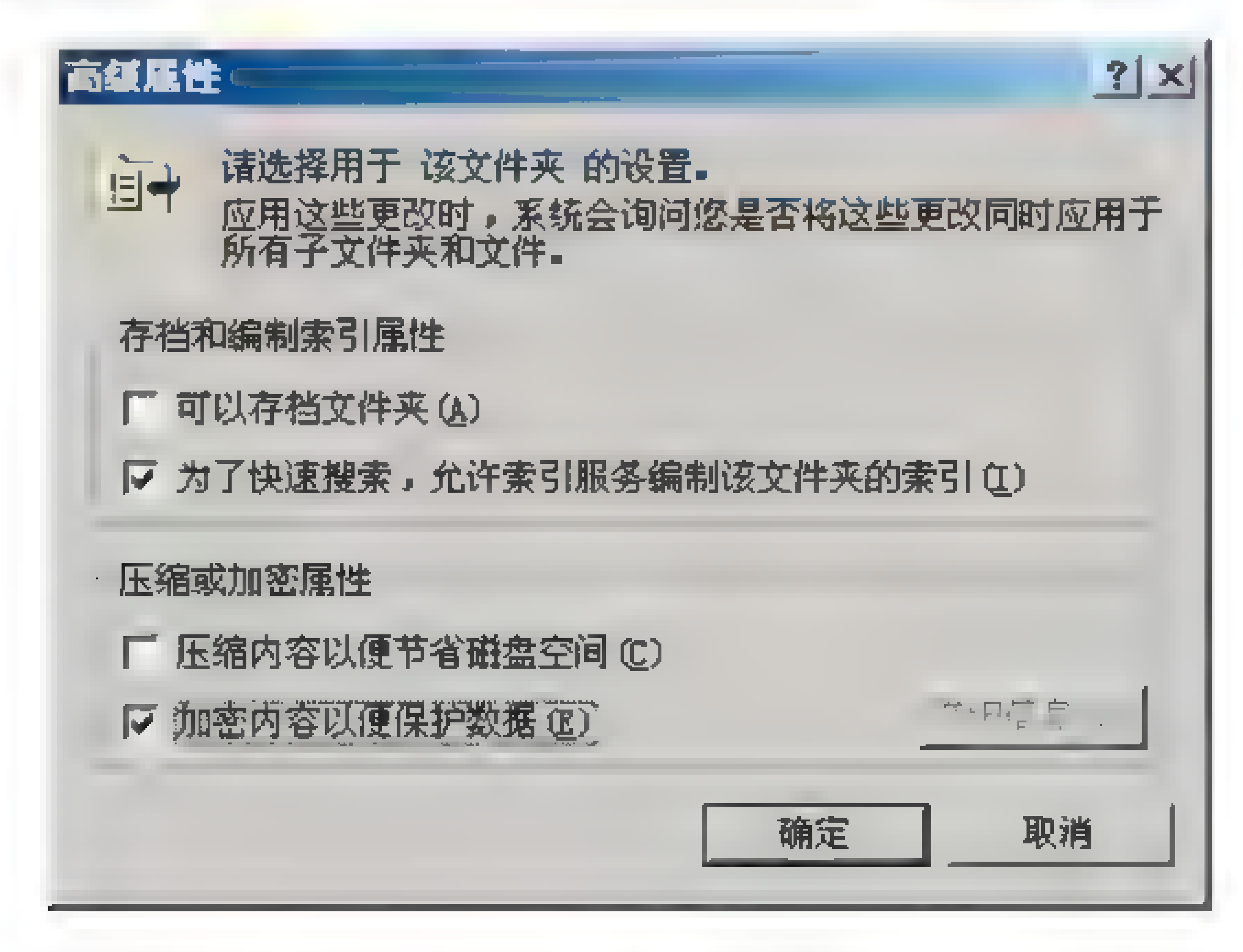


图 4-29

4.7.4　Windows Server 2003 安全管理采用的对策

管理员在管理 Windows Server 2003 时，一定要从物理安全、登录安全、用户安全、文件系统和打印机安全、注册表安全、RAS 安全、数据安全、各应用系统安全等方面制定强化安全的措施。Windows Server 2003 的安全管理的对策如下。

1．物理安全管理

- 去掉或锁死软盘驱动器，禁止 DOS 或其他操作系统访问 NTFS 分区。
- 在服务器上设置系统启动口令，设置 BIOS 禁用软盘引导系统。
- 不创建任何 DOS 分区。
- 保证机房的物理安全。
- 磁盘权限设置：C 盘只给 administrators 和 system 权限，其他的权限不给，其他的盘也可以这样设置。这里给的 system 权限也不一定需要给，只是由于某些第三方应用程序是以服务形式启动的，需要加上这个用户，否则造成不能启动。

2．及时更新补丁

用最新的 Service Pack 升级 Windows Server 2003，因为服务包包括所有补丁程序和后来发表的很多安全补丁程序。系统升级、打操作系统补丁，尤其是 IIS 6.0 补丁、SQL SP3a 补丁，甚至 IE 6.0 补丁也要打。同时及时跟踪最新漏洞补丁。

3．避免给用户定义特定的访问控制

将用户以“组”的方式进行管理是一个用户管理的有效方法。如果一个用户在公司里的角色变了，很难跟踪并更改他的访问权。为每个用户指定一个工作组，为工作组指定文件、文件夹访问权。如果要收回或更改某个用户的访问权，只要把该用户从工作组中删除或指定另一个工作组。

4．实施账号及口令策略

用域用户管理器配置口令策略，选择口令要注意以下几点：

- 登录名称中字符不要重复或循环。
- 至少包含两个字母字符和一个非字母字符。
- 至少有 6 个字符长度。
- 不要用用户的姓名，相关人物、著名人物的姓名，以及用户的生日和电话号码及其他容易猜测的字符组合等。
- 要求用户定期更改口令。
- 给系统的默认用户特别是 Administrator 改名并设置复杂密码。
- 不要使用无口令的账号，否则会给安全留下隐患。
- 禁用 Guest 账号。
- 设置账号锁定。

5. 控制远程访问服务

远程访问是入侵者攻击Windows Server 2003系统的常用手段，Windows Server 2003集成的防止外来入侵最好的功能是认证系统。Windows Server 2003客户机不仅可以交换加密用户ID和口令数据，而且还使用Windows专用的挑战/响应协议，这可以确保决不会多次出现相同的认证数据，还可以有效阻止内部黑客捕捉网络信息包。同时，如条件允许，应该使用回叫安全机制，并尽量采用数据加密技术，保证数据安全。对于个人用户来说，终端服务一般来说是不适用的，它的危险性远大于它带来的帮助，它允许从网络中的任何虚拟计算机上管理你的机器。使用运行 Windows Server 2003家族操作系统的任何计算机，通过管理远程桌面，来远程管理服务器。使用系统自带的远程桌面连接即可完成连接和控制。所以，对于普通用户来说，必须禁止终端服务。

6. 启动审核功能

为防止未经授权的访问，可以利用域用户管理器启用安全审核功能，以便在事件查看器安全日志中记录未经授权的访问企图，以便尽早发现安全漏洞。但要结合工作实际，设置合理的审计规则，切忌审查事件太多，以免没有时间全部审查安全问题。推荐的要审核的项目是：

- 登录事件。
- 账户登录事件。
- 系统事件。
- 策略更改。
- 对象访问。
- 目录服务访问。
- 特权使用。

具体的审核操作是：在“运行”对话框中的“打开”下拉列表框中输入 gpedit.msc，按回车键，打开组策略编辑器，选择“计算机配置”→“Windows 设置”→“安全设置”→“审核策略”。在创建审核项目时需要注意的是如果审核的项目太多，生成的事件也就越多，那么要想发现严重的事件也越难。当然，如果审核的太少也会影响你发现严重的事件，需要根据情况在这两者之间做出选择。

7. 确保注册表安全

首先，取消或限制对regedit.exe、regedit32.exe的访问；其次，利用regedit.exe或文件管理器设置只允许管理员访问注册表，其他任何用户不得访问注册表。

8. 应用系统的安全

在Windows Server 2003上运行的应用系统，如Web服务器、FTP服务器、E-mail服务器，Internet Explorer和IIS等，应及时通过各种途径（如Web站点）获得其补丁程序包，以解决其安全问题。把IIS中的sample、scripts、iisadmin和msadc等Web目录设置为禁止匿名访问并限制IP地址，把FTP、Telnet的TCP端口改为非标准端口。Web目录、CGI

目录、scripts 目录和 WinNT 目录等重要目录要用 NTFS 的特性设置详细的安全权限，包含注册表信息的 WinNT 目录只允许管理员完全控制。凡是与系统有关的重要文件，除了 Administrator，其他账号都应该设置为只读权限，而不是由 Everyone 完全控制。

9．取消 TCP/IP 上的 NetBIOS 绑定

Windows Server 2003 系统管理员可以通过构造目标站 NetBIOS 名与其 IP 地址之间的映像，对 Internet 或 Intranet 上的其他服务器进行管理，但非法用户也可从中找到可乘之机。如果这种远程管理不是必须的，就应该立即取消（通过网络属性的绑定选项，取消 NetBIOS 与 TCP/IP 之间的绑定）。如 NetBIOS 端口 139，要禁止这样的端口。对于个人用户来说，安装中默认的有些端口确实是没有什么必要的，关掉端口也就是关闭无用的服务。139 端口是 NetBIOS 协议所使用的端口，在安装了 TCP/IP 协议的同时，NetBIOS 也会被作为默认设置安装到系统中。139 端口的开放意味着硬盘可能会在网络中共享，网上黑客也可通过 NetBIOS 知道你的计算机中的一切。在 Windows Server 2003 中，关闭 139 端口的具体步骤如下。

（1）单击“开始”，选择“连接到”，然后打开所有连接，在本地连接处右击，在弹出的快捷菜单中选择“属性”命令，进入属性对话框，然后去掉“Microsoft 网络的文件和打印机共享”前面的选中标记，如图 4-30 所示。

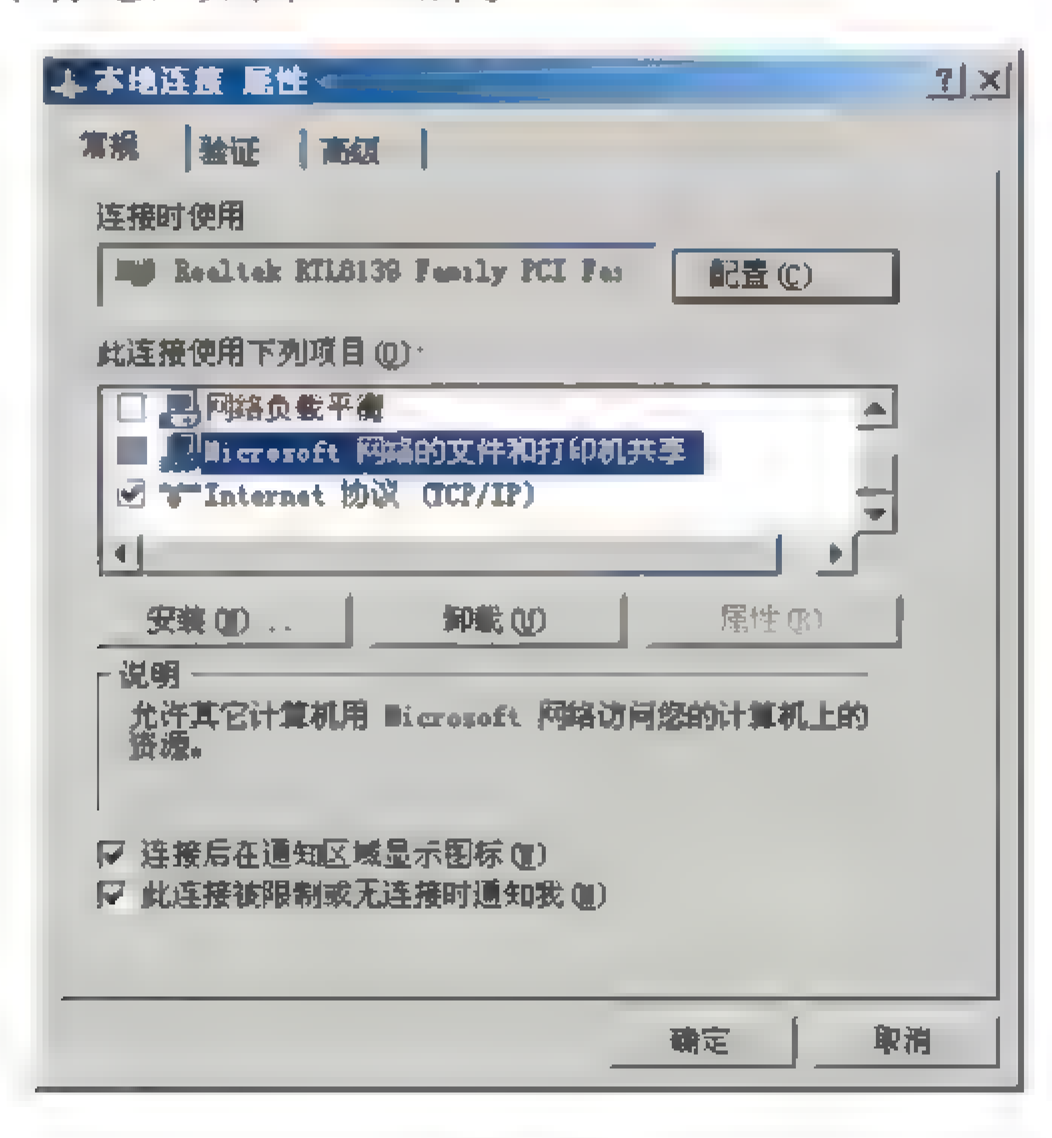


图 4-30

（2）双击 Internet 协议（TCP/IP），单击“高级”按钮，然后选择 WINS 选项卡，选中“禁用 TCP/IP 上的 NetBIOS”单选按钮，这样就禁用了 TCP/IP 上的 NetBIOS 绑定。如图 4-31 所示。

10．Internet Explorer 增强的安全配置

微软新一代的 Windows Server 2003 操作系统在安全性能方面得到了加强。比如在使用

Windows Server 2003 自带的 IE 浏览器浏览网页时，每次都会弹出一个安全提示框，是否启用 Internet Explorer 增强的安全配置对话框。

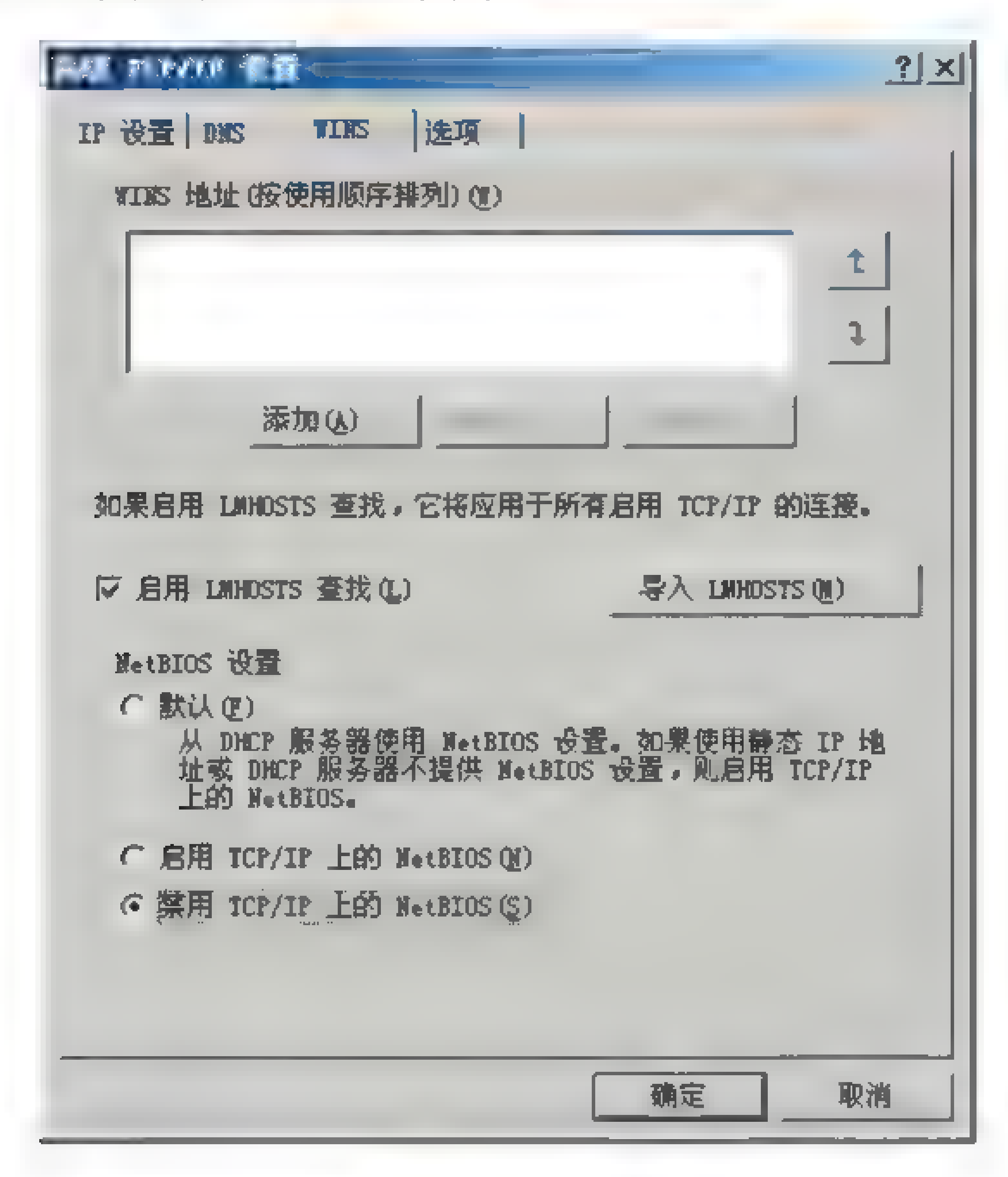


图 4-31

11．清除默认共享隐患

Windows Server 2003 系统在默认安装时会产生默认的共享文件夹。每个盘符都被 Windows 自动设置了共享，其共享名为盘符后面加一个符号 $（共享名称分别为 c$、d$、ipc$及 admin$）。也就是说，只要攻击者知道了该系统的管理员密码，就有可能通过“\\工作站名\共享名称”的方法，来打开系统的指定文件夹。所以需要将 Windows Server 2003 系统默认的共享隐患，立即从系统中清除掉。具体方法是编写脚本，然后把脚本文件放在 system32\GroupPolicy\User\Scripts\Logon 目录下，在组策略中打开用户配置，然后找到登录脚本。如图 4-32 所示，双击登录，将编写好的脚本添加即可，重新启动时，这个共享隐患就不存在了。

12．禁用 IPC 连接

IPC$（Internet Process Connection）是共享“命名管道”的资源，它是为了让进程间通信而开放的命名管道，通过提供可信任的用户名和口令，连接双方计算机即可以建立安全的通道并以此通道进行加密数据的交换，从而实现对远程计算机的访问。它是 Windows NT/2000/XP/2003 特有的功能，它有一个特点，即在同一时间内，两个 IP 之间只允许建立一个连接。NT/2000/XP/2003 在提供了 IPC$功能的同时，在初次安装系统时还打开了默认

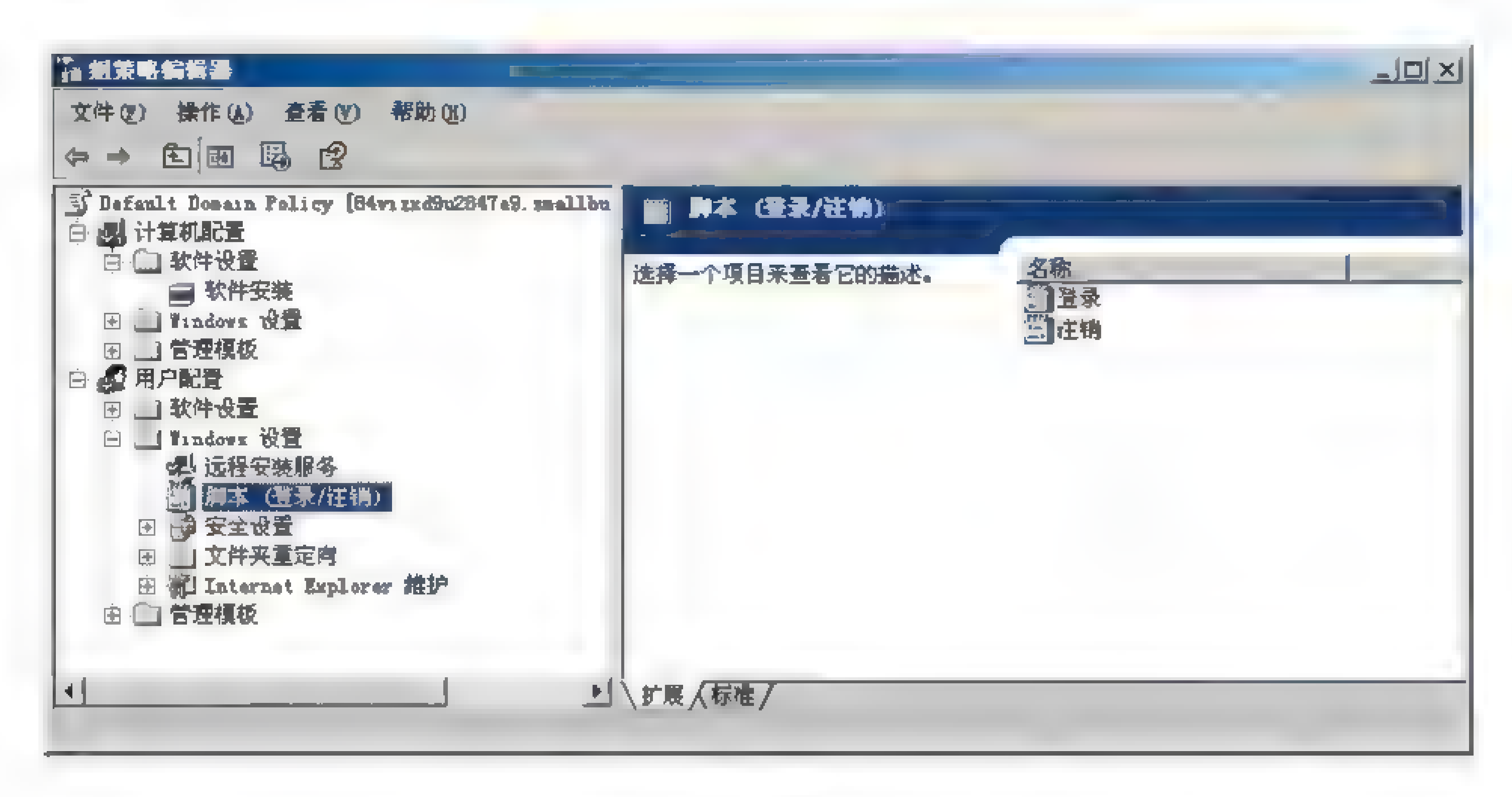


图　4-32

共享，即所有的逻辑共享(c$、d$、e$、…)和系统目录 winnt 或 Windows(admin$)共享。所有的这些，微软的初衷都是为了方便管理员的管理，但也为简称为 IPC 入侵者有意或无意地提供了方便条件，导致了系统安全性能的降低。在建立 IPC 的连接中不需要任何黑客工具，在命令行里输入相应的命令就可以了，不过有个前提条件，那就是需要知道远程主机的用户名和密码。打开 CMD 后输入如下命令即可进行连接：net use\\ip\ipc$ "password" /user:"usernqme"。可以通过修改注册表来禁用 IPC 连接。打开注册表编辑器。找到如下组建 HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa 中的 restrictanonymous 子键，将其值改为 1 即可禁用 IPC 连接。

13．清空远程可访问的注册表路径

Windows 2003 操作系统提供了注册表的远程访问功能，只有将远程可访问的注册表路径设置为空，才能有效地防止黑客利用扫描器通过远程注册表读取计算机的系统信息及其他信息。具体操作为：打开组策略编辑器，展开“计算机配置”→“Windows 设置”→“安全设置”→“本地策略”→“安全选项”，在右侧窗口中找到“网络访问：可远程访问的注册表路径”，然后在打开的窗口中，将可远程访问的注册表路径和子路径内容全部设置为空即可。如图 4-33 所示。

14．禁止不必要的服务，杜绝隐患

服务从本质上说也只是一个程序，它与其他程序所不同的地方在于它提供一种特殊的功能来支持系统完成特定的工作。Windows Server 2003 安装完后默认有 84 项服务，默认随系统启动的有 36 项。在 Windows Server 2003 发行后不到两个月就被人发现有了一个基于一项默认服务的漏洞，幸好这个漏洞对 Windows Server 2003 的危害程度较低，而且这项服务默认状态下是关闭的。具体操作为选择管理工具，然后单击打开服务，所有的服务都如图 4-34 所示。用户可以根据实际来启动和禁用相关的服务。

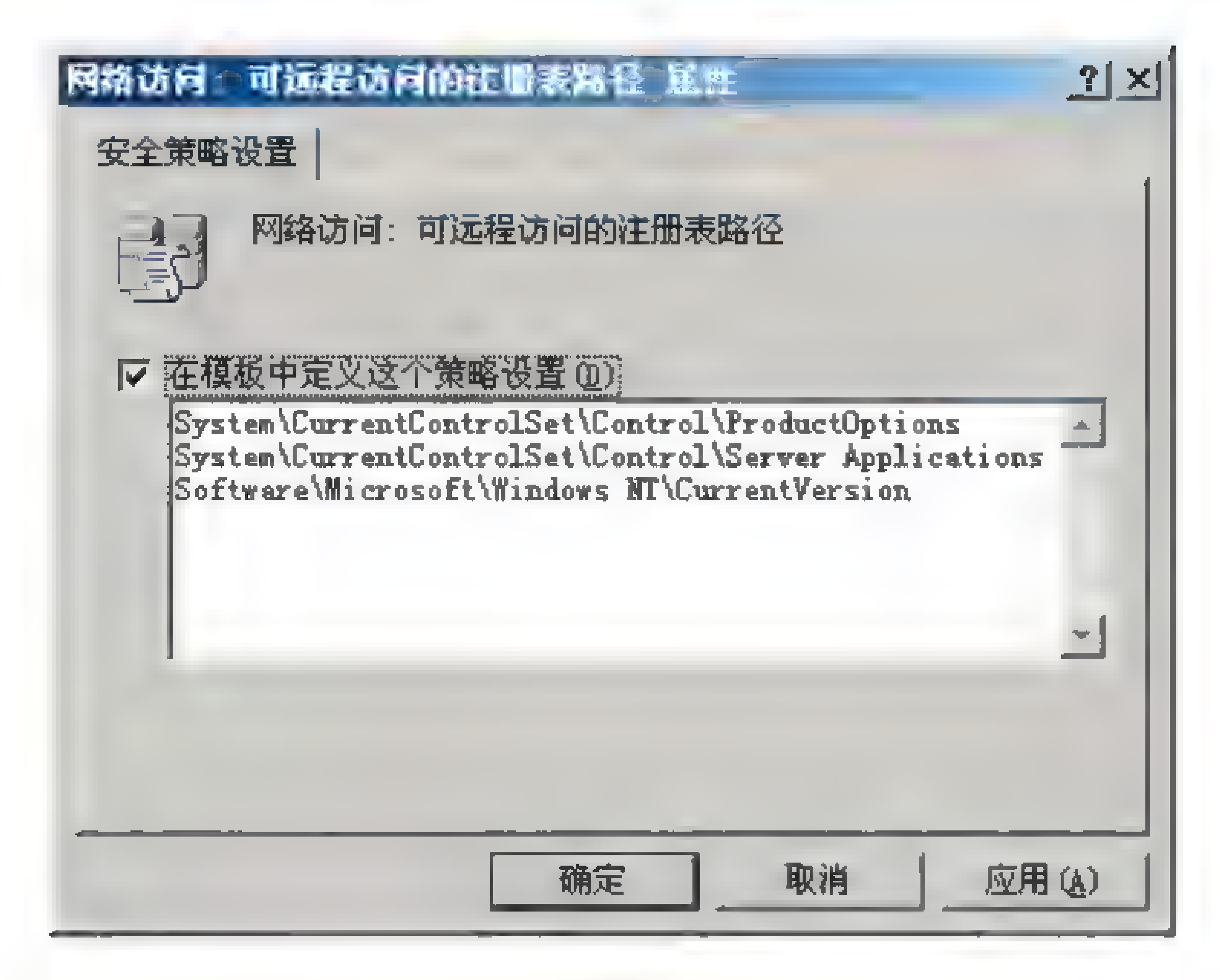


图 4-33

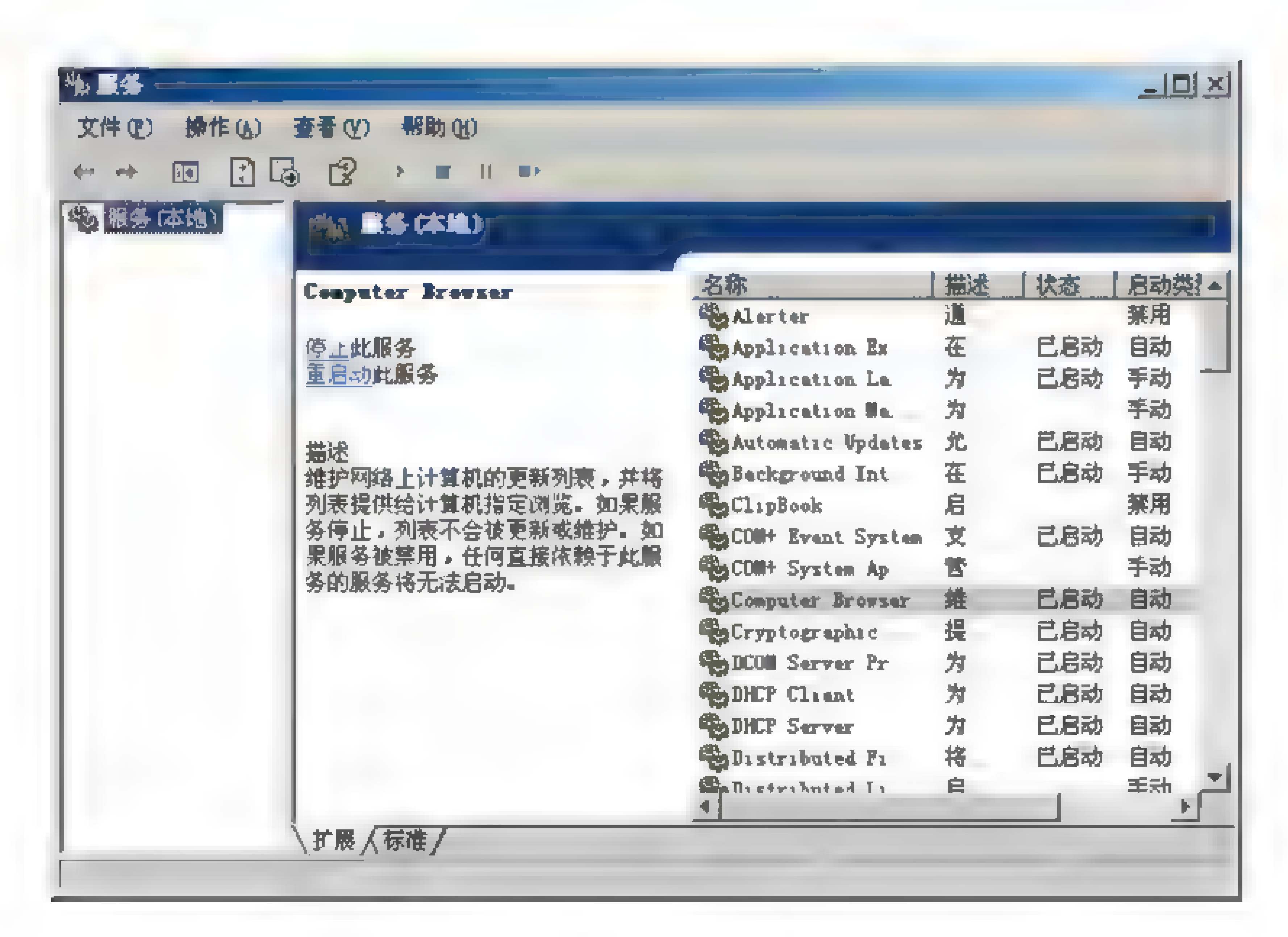


图 4-34

建议关闭的服务如下。

- Computer Browser：维护网络上计算机的最新列表及提供这个列表。
- Task scheduler：允许程序在指定时间运行。
- Messenger：传输客户端和服务器之间的 NET SEND 和警报器服务消息。

- Distributed File System：局域网管理共享文件，不需要禁用。
- Distributed linktracking client：用于局域网更新连接信息，不需要禁用。
- Error reporting service：禁止发送错误报告。
- Microsoft Search：提供快速的单词搜索，不需要可禁用。
- NTLMSecuritysupportprovide：telnet 服务和 Microsoft Search 用的，不需要禁用。
- PrintSpooler：如果没有打印机可禁用。
- Remote Registry：禁止远程修改注册表。
- Remote Desktop Help Session Manager：禁止远程协助。
- Workstation：如果关闭，远程 NET 命令列不出用户组。

把不必要的服务都禁用，尽管这些不一定能被攻击者利用得上，但是按照安全规则和标准来说，多余的东西就没必要开启，减少一份隐患。

总体来说，作为操作系统的安全管理员，应该随时警惕系统、安全和应用程序的日志，警惕注册表启动项的变化、警惕用户账户等敏感的地方是保证你的系统安全的必要条件。经常备份数据以应付灾难性事故。用户和权限的分配应当本着最小权限原则，即在不影响用户正常使用的情况下，为其分配最小的权限。感觉有时候也是很重要的，系统突然变慢就要先凭感觉判断一下是否感染了病毒等。系统安全是个长期性的工作，要时刻提高安全防范意识。

4.8　安 全 工 具

4.8.1　Nbtstat 实用命令

Nbtstat 命令是一个端口扫描程序，它能扫描目标机或网络的端口信息，如果端口扫描程序报告端口 139 在目标机或网络上是开放的，那么使用 Nbtstat 命令，可以查询网络机器上的 NetBIOS 信息。同时 Nbtstat 在进行安全检查时也经常用到。下面介绍一下这个命令。

1．Nbtstat 命令格式

Nbtstat[-aRemoteName][-AIP_address][-c][-n][-R][-r][-S][-s][iNTerval]

参数说明如下。

- -a：列出为其主机名提供的远程计算机名字表。
- -A：列出为其 IP 地址提供的远程计算机名字表。
- -c：列出包括了 IP 地址的远程名字高速缓存器。
- -n：列出本地 NetBIOS 名字。
- -r：列出通过广播和 WINS 解析的名字。
- -R：消除和重新加载远程高速缓存器名字表。
- -S：列出有目的地 IP 地址的会话表。
- -s：列出会话表对话。

2. Nbtstat 生成的列标题的含义

- Input：接收到的字节数。
- Output：发出的字节数。
- In/Out：是从计算机（出站）还是从另一个系统连接到本地计算机（入站）。
- Life：在计算机消除名字表高速缓存表目前“度过”的时间。
- LocalName：为连接提供的本地 NetBIOS 名字。
- RemoteHost：远程主机的名字或 IP 地址。
- Type：一个名字可以具备两个类型之一。
  unique 或 orgroup 类型在 16 个字符的 NetBIOS 名中，最后一个字节往往有具体含义，因为同一个名可以在同一台计算机上出现多次。这表明该名字的最后一个字节被转换成了十六进制。
- StateNetBIOS 连接将在下列“状态”（任何一个）中显示。

状态含义如下。

- Accepting：进入连接正在进行中。
- Associated：连接的端点已经建立，计算机已经与 IP 地址联系起来。
- Connected：这是一个好的状态，表明已被连接到远程资源上。
- Connecting：你的会话试着解析目的地资源的名字——IP 地址映射。
- Disconnected：计算机请求断开，并等待远程计算机作出这样的反应。
- Disconnecting：连接正在结束。
- Idle：远程计算机在当前会话中已经打开，但现在没有接受连接。
- Inbound：入站会话试着连接。
- Listening：远程计算机可用。
- Outbound：你的会话正在建立 TCP 连接。
- Reconnecting：如果第一次连接失败，就会显示这个状态，表示试着重新连接。

下面是一台机器的 nbtstat 反应样本：

```
C:\>nbtstat C A x.x.x.x
NetBIOS Remote Machine NameT able
Name   Type    Status
DATARAT<00>UNIQUERegistered
R9LABS<00>GROUPRegistered
DATARAT<20>UNIQUERegistered
DATARAT<03>UNIQUERegistered
GHOST<03>UNIFQUERegistered
DATARAT<01>UNIQUERegistered
MACAddress=00-00-00-00-00-00
```

通过上表能知道该计算机的如下信息。

名称编号类型的使用

- 00U　　　　工作站服务
- 01U　　　　邮件服务
- _MSBROWSE_01G　　　　主浏览器

- 03U 邮件服务
- 06U RAS 服务器服务
- 1FU NetDDE 服务
- 20U 文件服务器服务
- 21U RAS 客户机服务
- 22U ExchangeINTerchange
- 23U ExchangeStore
- 24U ExchangeDirectory
- 30U 调制解调器共享服务器服务
- 31U 调制解调器共享客户机服务
- 43U SMS 客户机远程控制
- 44U SMS 管理远程控制工具
- 45U SMS 客户机远程聊天
- 46U SMS 客户机远程传输
- 4CU DECPathworksTCP/IP 服务
- 52U DECPathworksTCP/IP 服务
- 87U ExchangeMTA
- 6AU ExchangeIMC
- BEU 网络监控代理
- BFU 网络监控应用
- 03U 邮件服务
- 00G 域名
- 1BU 域主浏览器
- 1CG 域控制器
- 1DU 主浏览器
- 1EG 浏览器服务选择
- 1CG INTernet 信息服务器
- 00U INTernet 信息服务器
- [2B]U LotusNotes 服务器
- IRISMULTICAST[2F]G LotusNotes
- IRISNAMESERVER[33]G LotusNotes
- Forte $ND800ZA[20]U DCAIrmalan 网关服务
- Unique(U) 该名字可能只有一个分配给它的 IP 地址。在网络设备上，一个要注册的名字可以出现多次，但其后缀是唯一的，从而使整个名字是唯一的
- Group(G) 一个正常的群，一个名字可以有很多个 IP 地址
- Multihomed(M) 该名字是唯一的，但由于在同一台计算机上有多个网络接口，这个配置允许注册。这些地址的最大编号是 25
- INTernetGroup(I) 这是用来管理 WinNT 域名的组名字的特殊配置

- DomainName(D)NT　　　　　　　提供的新内容

网络入侵者可以使用 nbtstat 获取目标机中的信息。根据这些信息，网络入侵者就能攻击相关的服务或软件包，随后通过远程机收集可能的用户名。网络登录包括两个部分：用户名和口令。一旦网络入侵者掌握了有效的用户列表，他就能获得一半的有效登录信息。现在采用了 nbtstat 命令，网络入侵者就能掌握从本地注册到该台机器上的任何人的登录名。在通过 nbtstat 命令得到的结果中，采用<03>识别符的表目是用户名或机器名。另外，还可以通过空 IPC 会话和 SID 工具来收集用户名。

IPC$（进程间通信）共享是 SERVER 2003 主机上一个标准的隐藏共享，主要用于服务器到服务器的通信。NT 主机用来互相连接并通过这个共享获得各种必要的信息。鉴于在各种操作系统中都有很多设计特征，网络入侵者已经懂得利用这种特征来达到他们的目的。通过连接这个共享，网络入侵者从技术上就能够实现与你的服务器的有效连接。通过与这个共享的空连接，网络入侵者就能够在不需要提供任何身份证明的情况下建立这一连接。

要与 IPC$共享进行空连接，网络入侵者就在命令提示符下发出如下命令：

```
c:\>netuse\\[目标主机的IP地址]\ipc$""/user:""
```

如果连接成功，网络入侵者就会有很多事情可做，不只是收集用户列表，不过他是以收集用户列表开始的。

4.8.2　Netview

如果有了空 IPC 会话，网络入侵者也能获得网络共享列表，否则就无法得到。为此，网络入侵者希望了解到在你的机器上有哪些可用的网络共享。为了收集到这些信息，要采用下列这个标准的 netview 命令：

```
c:\>netview\\[远程主机的IP地址]
```

根据目标机的安全约束规则，可以拒绝或不拒绝这个列表。举例如下：

```
C:\>netview\\0.0.0.0
System error5 has occurred.
Access is denied.
C:\>netuse\\0.0.0.0\ipc$""/user:""
The command completed successfully.
C:\>netview\\0.0.0.0
Shared resource sat\\0.0.0.0
Sharename type use dascomment
Accelerator Disk AgeNT Accelerator share for Seagate backup.
InetpubDisk mircDisk
NETLOGON Disk Logon server share.
www pages Disk
The command completed successfully.
```

4.8.3 Usersat

这个命令行实用程序显示特定域中各个用户的用户名、全名及最后一次登录的日期和时间。下面是根据远程网络通过一个空 IPC 会话采用这个工具进行的实际剪切和粘贴。

```
C:\NTRESKIT>usrstatdomain4
Usersat\\STUDENT4
Administrator--logon:TueNov1708:15:251998
Guest--logon:MonNov1612:54:041998
IUSR_STUDENT4-INTernetGuestAccouNT-logon:MonNov1615:19:2619
98
IWAM_STUDENT4-WebApplicationManageraccouNT-logon:Never
laurel--logon:Never
megan--logon:Never
```

现在说明一下，在真正的攻击发生前，把一个映射放到通过#PRE/#DOM 标记映射 StudeNT4 机器及其域活动状态的 lmhosts 文件中(下面详述)。然后把表目预加载到 NetBIOS 高速缓存器中，同时建立一个空 IPC 会话。这个命令是根据域名发出的。最后，该工具会向主域控制器查询这个域。

4.8.4 Global

这个命令行实用程序显示远程服务器或域上全局群组的成员。如上所述，这个实用程序是与 Lmhosts/IPC 映射一起使用的。下面是 global 工具的实际俘获。在这个例子中，“域用户”是 Windows NT 域中出现的标准默认全局群组。在此采用这个工具向 Domain1 查询“域用户”群组中所有用户的列表。

```
C:\>global"DomainUsers"domain1
Bob
SPUPPY$
BILLYBOB$
Bill
IUSR_BILLYBOB
IWAM_BILLYBOB
IUSR_SPUPPY
IWAM_SPUPPY
```

4.8.5 local 工具

local 工具像 global 工具一样操作，不同之处是，它向机器查询本地群组的成员，而不是全局群组的成员。下面是 local 工具向服务器查询其管理员群组列表的例子。

```
C:\>local "administrators" domain1
Bob
DomainAdmins
Bill
```

4.8.6 NetDom 工具

NetDom 工具是一个向服务器查询它在域中的角色及向机器查询其 PDC 的工具。另外，NetDom 工具还与 Lmhosts/IPC 映射协同工作。下面是该工具获得的信息及其标准输出：

```
Queryingdomaininformationoncomputer\\SPUPPY…
Thecomputer\\SPUPPYisadomaincoNTrollerofDOMAIN4.
SearchingPDCfordomainDOMAIN4…
FoundPDC\\SPUPPY
Thecomputer\\SPUPPYisthePDCofDOMAIN4.
```

4.8.7 NetWatch 工具

NetWatch 工具是一个向调用该工具的用户提供远程机上的共享列表的工具。同样这个工具也能与 Lmhosts/IPC 映射一起使用。这个工具的缺点是，人们能够利用该工具来索取远程机上的隐藏共享列表。

4.8.8 Netusex

一旦攻击者掌握了远程共享列表，他就会试着映射到远程共享。这一攻击的命令结构如下。

Netusex 是一个能映射到远程共享的攻击性命令,它的格式如下：

```
c:\>netusex:\\0.0.0.0\inetpub
```

注意：这种攻击发生的条件是共享不设密码或权限分配为 everyone 时。

攻击都把网络映射到你的计算机上，入侵者并不仅仅是把驱动器映射到通过 netview 命令显示出来的共享上，还可以入侵 Windows NT 隐藏的管理共享，Windows NT 为该机器上的每一个驱动器都创建 IPC$共享和一个隐藏共享（即一台有 C、D 和 E 驱动器的机器会有对应的 C$、D$和 E$的隐藏共享）。除此之外，还可以直接映射到 NT 安装路径的隐藏 ADMIN$共享（如果把 NT 安装在 C:\winNT 目录下，ADMIN$就映射到该驱动器的确切位置）。网络入侵者得到一个用户的权限，那么他就可以得到整个内部网络的权限，所以共享攻击也会很危险。

第 5 章

Linux 网络操作系统的安全管理

作为自由软件，Linux 不管是在客户机端还是在服务器端都得到了广泛的应用。与其他操作系统相比，Linux 在许多方面都具有自己的优势，管理上也具有自己的特色。为此，本章使用较大的篇幅，较为全面和完整地介绍 Linux 网络操作系统的安全管理方法。

5.1 系统安全

Linux 系统的安全性是有目共睹的，因为它是一个开放性的操作环境，正是因为代码的开放性，一旦系统有漏洞，则相应的补丁马上出台，在这一章中将详细讲解基于 Linux 操作系统的安全管理。

5.1.1 C1/C2 安全级设计框架

一般来说，系统安全性总是至少包含两种类型的安全性问题，一种问题是设计上的，比如对于 DOS/Windows，因为它们的设计就是非安全的，系统设计上不提供任何保护的功能。另一种问题是实现上的，就是实际的程序设计有问题，导致系统存在缺陷或者后门。下面讲述操作系统的安全性设计。

1985 年，美国国防部公布了一个《DoD 可信计算机系统评估标准》（Trusted Computer System Evaluation Criteria，TCSEC）。这个标准成为一段时间内评价计算机系统的设计安全性的标准。虽然实际上这个标准有点大而无用，但是对于我们讨论基本问题是有帮助的，而且确实有一些系统（当然不是 Linux）是按照这个标准去实现的，因此先简单地介绍一下这个框架的概念。

TCSEC（TDI）将系统划分为 4 组（division）7 个等级，依次是 D；C（C1，C2）；B（B1，B2，B3）；A（A1），安全级别从左至右逐步提高，各级之间向下兼容，也就是说高级别必须拥有低级别的一切特性。D 级的定义最低，简单地说就是没有任何安全性，可以随便破坏系统而且绝不会有限制。对我们来说，比较重要的是 C 类的两个级别和 B 类的两个级别。这些级别的简单描述如表 5-1 所示。

事实上，C2 类是安全性系统的基本要求，也是绝大部分系统的标准要求，UNIX/Linux

系统也不例外，而真正意义上的安全系统普遍要求达到B级。不过，必须说明的是，操作系统通过B1级认证的并不多，而通过B2级认证的更是稀少。这是因为安全认证是专门机构做的事情，经常会做上几年，因此几乎任何真正的商业系统都不会有时间去做这种认证。但是即使不考虑证书问题而是简单地按照上面说的规则，你也会发现达到B2级安全程度的系统是极少见的。其原因一方面是这样的操作系统设计起来比较麻烦，另一方面是，这样设计的系统，使用上很不方便。毕竟大部分操作系统和硬件都要把用户的方便放在第一位。对于易用性和可靠性的折中的结果，大部分常用系统属于C2等级或者经过修改后可以达到C2等级（不过可能有个别C2要求鉴于系统运行效率和易用性的要求被取消了）。

表5-1　安全级别描述简表

类　别	等　级	描　　述
C	C1	所有的用户都被分组； 对于每个用户，必须登记后才能使用系统； 系统必须记录每个用户的登记； 系统必须对可能破坏自身的操作发出警告
	C2	在C1的基础上增加以下几条要求： 所有的对象都有且仅有一个物主； 对于每个试图访问对象的操作，都必须检验权限，对于不符合权限要求的访问，必须予以拒绝； 有且仅有物主和物主指定的用户可以更改权限； 管理员可以取得对象的所有权，但不能归还； 系统必须保证自身不能被管理员以外的用户改变； 系统必须有能力对所有的操作进行记录，并且只有管理员和由管理员指定的用户可以访问该记录
B	B1	在C2的基础上增加以下几条要求： 不同的组成员不能访问对方创建的对象，但管理员许可的除外； 管理员不能取得对象的所有权
	B2	在B1的基础上增加以下几条要求： 所有的用户都被授予一个安全等级； 安全等级较低的用户不能访问高等级用户创建的对象

下面简单地介绍一下C1/C2的要求。它包括三个基本的部分：

（1）身份认证，每个用户都必须在系统中标志其身份。

（2）系统的资源（文件、内存等）被归于不同的所有者，对这些资源的访问必须验证用户权限。

（3）系统要对用户的行动进行记录。

其中，用户和权限的管理是至关重要的，下面逐个来讨论这些问题。

5.1.2　身份认证

身份认证是用户进入系统的第一步。在Linux设计框架里面，用户的身份和权限是分开的，每个用户的身份并不影响它对于特定物件的权力——除了超级用户以外。这使得用户身份的认证比较简单，但是也带来了一定的问题，比如随便提升用户权限的问题等。在

本节中，首先考虑身份认证系统。

身份认证系统最基本的实现是 Linux login 程序，不过其他各种应用程序也一样要通过身份认证来确定用户身份。要注意事实上身份认证仅仅是个程序实现，决定用户身份，是它启动应用程序的 uid，因此 Linux 下的身份认证其实只是判断用户并且授予它一个合法的 uid 的过程。而授予 uid 的过程却是超级用户（或者超级用户进程）的工作，事实上授予程序只要愿意，随便怎么做都可以，因此并没有完全统一的办法来实现身份认证。在 Linux 中传统上采用一些约定俗成的方法来认证用户身份，其中包括最基本的 password/shadow 体系和 PAM 体系。在这些体系的基础上，程序可以用系统调用来完成对用户的认证，但是，如果系统进程非要抛开调用，用自己的一套方法进行用户认证，Linux 也不会禁止。

1．Shadow 身份认证体系

最基本的身份认证系统由口令验证构成。用户输入一个口令，与系统进行比较，如果合法，就可以进入系统。显然，最重要的是验证口令与这个用户的预设口令是否相同。当然，口令不是用明文进行比较，而是采用下面的步骤。

（1）系统记录用户的原始口令，并将其加密保存在系统中，口令原文则被丢弃。

（2）当用户登录系统的时候，输入口令，系统用同样的加密算法将用户输入的口令转换成密文。

（3）比较保存的密文和现在得到的密文，如果相同，允许用户登录系统。

口令加密可以使用一般的对称密钥算法，但是 Linux 普遍使用类似 MD5 一类的 HASH 方法。这类算法是不对称的，也就是说从密文（或者说校验和）是不可能知道原文的。

尽管如此，如果看到了加密结果，用户仍然可以用穷举法来获得密码，因为标准的 UNIX 口令体系只有 8 位。如果考虑到大部分人喜欢使用小写字母和很短的密码的话，那么，穷举所有的字符组合构成试探口令，然后执行第（2）步，对于比较短的密码是有可能搜索到正确的口令字符串的。

即使对于长口令（8 位甚至超过 8 位）也不是不可能穷举，这是因为许多人有用特定单词/短语的组合作为口令的习惯。标准的方法是构造一个巨大的词典（可以来自生日、姓名和常用词汇表等），然后从词典中选出词汇进行组合形成试探密码，并用上述方法试验，直至成功。

因为这种穷举试探方法的存在，所以允许一般用户看到系统记录的密码密文是危险的。但是对于 Linux，/etc/passwd 文件必须对于所有用户是可读的，因此 Linux 系统采用所谓 shadow 的方法，即把口令认证文件分成两个，分别是/etc/passwd 和/etc/shadow。/etc/passwd 文件包含了“公共”的，也就是一般用户可以访问的内容；而/etc/shadow 文件包含了加密的密文，其内容只有超级用户可以访问。

这个方法尽管很简单，但是却给某些程序的身份认证带来了不必要的麻烦。例如邮件服务程序为了安全性有时候需要使用 chroot 方式，从而不能直接访问/etc 目录下面的文件，因此每次增加用户都需要修改系统。除此之外，这种直接文件访问的方法效率和可靠性都比较低，还存在被用缓冲区溢出等手段攻破的可能。

2．PAM 身份认证体系

由于这类问题，现代的 Linux 系统从 Sun 引入了 PAM 系统（可插拔认证模块）。必须

指出的是，只有在程序编写时选择了PAM库支持的时候，才能使用PAM认证。在这种情况下，程序调用PAM运行库，而运行库则根据当前的PAM系统管理设定来进行具体的认证过程，使得整个认证过程可以添加或者删除特定的功能，从系统核心中分离出来。

PAM认证部分是由一组模块构成的，相互可以堆叠。堆叠的意思就是说，可以连续执行多个模块或者让一个模块多次使用。因此，可以在一个PAM认证过程中使用多种认证模块，后面的认证过程的执行依赖于前面的认证模块结果。另外，还可以随时加入新的模块，加入之后，PAM客户程序无须重新编译，也无须做任何修改就可以使用这个新模块。

1）PAM系统的配置文件

PAM系统的配置文件，按照具体实现，可以分为/etc/pam.conf文件和/etc/pam.d/目录两种方式，Linux一般使用第二种方式。这种方式是，在/etc/pam.d/目录下，存放着一些分离的配置文件，每个文件的文件名是这个文件控制的服务（telnet、pop和ftp等），而文件的内容是这个服务的PAM配置。一般来说，服务类型就是存取该服务的程序的名字，而不是提供服务的程序。例如下面是用ls命令列出的程序名（即服务）列表。

```
[wly@ cs pam.d] $ls
apacheconf      kbdrate         printtool                     screen
authconfig      kde             reboot                        serviceconf
authconfig-gtk  kisdndock       redhat-config-apache          smtp
bindconf        kpackage        redhat-config-bind            sshd
chfn            kppp            redhat-config-date            su
chsh            kscreensaver    redhat-config-network         sudo
cups            kuser           redhat-config-network-cmd     system-auth
dateconfig      kwuftpd         redhat-config-network-druid timetool
firewall-config locale-config   redhat-config-printer-gui     up2date
ftp             login           redhat-config-printer-tui     up2date-config
gdm             neat            redhat-config-services        up2date-nox
gdmconfig       other           redhat-config-time            v4l-conf
gnome-lokkit    passwd          redhat-config-users           xdm
gnorpm-auth     pop             rexec                         xscreensaver
halt            poweroff        rhn_register                  xserver
hwbrowser       ppp             rlogin
imap            printconf-gui   rsh
internet-druid  printconf-tui   samba
```

2）配置文件的内容

每个程序配置文件的内容大致如下（使用cat命令，文件名为login）。

```
[wly@ cs pam.d] $cat login
#%PAM-1.0
auth        required    /lib/security/pam_securetty.so
auth        required    /lib/security/pam_stack.so service=system-auth
auth        required    /lib/security/pam_nologin.so
account     required    /lib/security/pam_stack.so service=system-auth
```

```
password    required    /lib/security/pam stack.so service=system-auth
session     required    /lib/security/pam stack.so service=system-auth
session     optional    /lib/security/pam console.so
```

其中，以#开头的是注释行，其他行的格式是：

```
module-type control-flag module-path arguments
```

各参数解释如下。

- module-type：说明该行属于哪个模块，或者用在认证的哪个部分，取值如表 5-2 所示。

表 5-2　module-type 取值表

模块取值	描　述
auth	提供实际的认证过程，可能是提示输入并检查口令，设置保密字等
account	负责检查并确认是否可以进行认证，例如，账户是否到期，用户此刻是否可以登录等
password	用来设置口令
session	一旦用户认证通过，此模块将被用于用户使用其账户前的初始化工作，如加载用户根目录或开通用户的电子邮箱

- control-flag：用来设置本行认证过程成功或者失败后如何响应。因为 PAM 是完整的，客户程序无法看到 PAM 内部的东西，只能等待 PAM 完全执行之后查看最终的结果。为此，对于多模块认证，必须小心地设置认证响应，它的取值如表 5-3 所示。

表 5-3　control-flag 取值表

取　值	描　述
required	表示本模块必须返回成功才能通过认证，但是若该模块返回失败，其结果也不会立即通知用户，而是要等到同一 stack 中的所有模块全部执行完毕再将失败结果返回给应用程序，这样做是为了不让用户知道被哪个模块拒绝
requisite	和 required 类似，区别在于如果模块对用户的验证失败，PAM 马上返回一个错误信息，把控制权交回应用程序，不再执行其他模块进行验证
sufficient	表示如果一个用户通过这个模块的验证，PAM 结构就立刻返回验证成功信息，把控制权交回应用程序，后面的层叠模块即使使用 requisite 或者 required 控制标志，也不再执行；如果验证失败，这个模块将被忽略而执行下一个验证模块
optional	表示即使本行指定的模块验证失败，也允许用户享受应用程序提供的服务。使用这个标志，PAM 框架会忽略这个模块产生的验证错误，继续顺序执行下一层叠模块

- module-path：用来指明本模块对应的程序文件的路径名，一般采用绝对路径，如果没有给出绝对路径，默认该文件在目录/usr/lib/security 下。
- arguments：用来传递给该模块的参数。一般来说每个模块的参数都不相同，可以由该模块的开发者自己定义，但是也有几个共同的参数，如表 5-4 所示。

由于模块在不断地增加和更新，也可以由用户自行设计，因此没有完整的说明。但是在通常情况下，Linux 中总是有几个常用模块，如表 5-5 所示。

表 5-4 arguments 共同参数表

参数	描述
debug	该模块应当用 syslog()将调试信息写入系统日志文件中
no_warn	表明该模块不应把警告信息发送给应用程序
use_first_pass	表明该模块不能提示用户输入密码，而应使用前一个模块从用户那里得到的密码
try_first_pass	表明该模块首先应当使用前一个模块从用户那里得到的密码，如果该密码验证通不过，再提示用户输入新的密码
use_mapped_pass	该模块不能提示用户输入密码，而是使用映射过的密码
expose_account	允许该模块显示用户的账户名等信息，一般只能在安全的环境下使用，因为泄漏用户名会对安全造成一定程度的威胁

表 5-5 Linux 常用模块表

名称	描述
pam_stack	实现一个 PAM 堆栈，简单地说是调用另一个完整的 PAM 配置，这种调用利用 service=xxx 参数实现，xxx 是/etc/pam.d 中的 PAM 配置模块名字。例如，在前面的例子中，/lib/security/pam _ stack.so service=system-auth 表示要引用/etc/pam.d/system-auth 配置文件，而 pam_stack 行的成功或失败将决定于整个/etc/pam.d/system-auth 的成功或失败
pam_access	利用/etc/security/access.conf 文件对超级用户登录进行控制
pam_chroot	提供 chroot 功能
pam_cracklib	对密码的强度进行一定的检查，使用库文件 libcrack 和字典文件/usr/lib/crack-lib_dict
pam_deny	总是无条件地使认证失败
pam_env	设置或取消环境变量，配置文件为/etc/security/pam_env.conf
pam_filter	对输入输出流进行过滤
pam_ftp.so	对匿名 ftp 用户进行认证
pam_group	当用户在指定的终端上时赋予该用户相应的组权限
pam_issue	在提示用户输入用户名之前显示/etc/issue 文件的内容
pam_krb4	对用户密码进行 Kerberos 认证
pam_lastlog	在用户登录成功后显示用户上次登录的信息，并维护/var/log/lastlog 文件
pam_limits	利用/etc/security/limits.conf 限制用户会话所能使用的系统资源
pam_listfile	根据指定的某个文件决定是否允许或禁止提供服务
pam_mail	检查用户的邮箱中是否有新邮件
pam_mkhomedir	为用户建立主目录，模板目录在/etc/skel/
pam_motd	显示/etc/motd 文件的内容
pam_nologin	根据/etc/nologin 文件的存在与否来决定是否允许用户认证
pam_permit	总是无条件地使认证成功
pam_pwdb	使用 Password Database 进行认证，配置文件在/etc/pwdb.conf
pam_radius	提供远程身份验证拨入用户服务（RADIUS）的认证
pam_rhosts_auth	利用文件~/.rhosts 和/etc/hosts.equiv 对用户进行认证
pam_rootok	检查用户是否为超级用户，如果是超级用户则无条件地通过认证
pam_securetty	提供标准的 UNIX securetty 检查，检查方式的配置文件为/etc/securetty

续表

名 称	描 述
pam_time	提供基于时间的控制，比如限制用户只能在某个时间段内才能登录，配置文件在/etc/security/time.conf
pam_unix	进行标准的 UNIX 认证（使用/etc/passwd 和/etc/shadow）
pam_userdb	利用 Berkeley DB 数据库来检查用户/密码
pam_warn	利用 syslog 记录一条警告信息
pam_wheel	只允许 wheel 组的用户使用 su 命令切换到超级用户

按照上面所说的，查看刚才的 login 配置文件，可以看到，诸如限制超级用户登录和使用 nologin 文件等都是用模块实现的，因此可以很容易地增加和撤销。相应地，对于这些功能的控制，都需要修改这个配置文件。

PAM 中定义了一个特殊的服务类型 OTHER，它代表除了已经有了控制文件的服务之外的所有服务。设置这个服务类型对应的 PAM 文件，将会直接对所有未控制服务生效。

5.1.3 用户权限和超级用户

Linux 的基本用户隔离和存取授权功能是用文件权限实现的，使用标准的 UNIX 文件权限体系，每个文件有一个属主用户（user）和一个属主程序组（group），除此之外的用户都作为其他用户（other）。这样，每个文件存在三种存取权限：用户存取权限、组存取权限和其他用户的存取权限。

1．文件属性标志 drwx 及用户权限

另外，还有一个属性字用于标志文件属性，包括设备、文件、目录或是链接。每个存取权限由读、写、执行和用于执行文件的特殊标志构成，例如：

```
[root@ cs root]#ls -l
total 11482
drwxrwxr-x  4  root  wheel  792      Feb 5 2002   Darwin4.0-Linux
-rw-r--r--  1  wly   wly    10799692 Jul 21 08:52 Darwin4.0-Linux.tar.gz
drwx------  3  root  root   208      Mar 15 2002  Desktop
-rw-------  1  root  root   60976    May 27 09:19 mbox
drwx------  5  root  bin    240      Feb 17 2002  ncftpd-2.7.1
-rw-r--r--  1  root  root   887285   Feb 25 2002  ncftpd-2.7.1-linux-
                                                  x86-ex-port.tar.gz
-rw-r--r--  1  root  root   848      Jul 2 17:51  wget-log
```

最左面的一位是文件属性，然后每三个字符为一组，分别代表用户、组和其他用户的存取权限。例如：

```
drwxrwxr-x 4 root  wheel  792 Feb  5  2002 Darwin4.0-Linux
```

第一位字符为 d，表示这是一个目录，然后头三个字符 rwx 表示用户的拥有者可以对

它进行读、写和执行的操作，- 表示均不可以。同样，同组用户也拥有读写和执行权限，而其他用户只能读取或者执行，其中的减号表示写入位没有设置，因此其他用户不能写入这个文件。接下来，这个目录的所有者是root，而拥有这个目录的程序组是wheel。后者意味着，所有属于wheel用户组的用户都自动具有对这个文件的组访问权限：读、写和执行。这些权限只能被文件的属主（user）修改。

用户权限也可以用数字表示，即把每组的三个权限位翻译成数字：r=4，w=2，x=1，然后进行按位加（例如，r-x 为 5），最后将三个数连到一起。如上例中的 ncftpd-2.7.1-linux-x86-export.tar.gz 文件的权限数值就是0644。

2. 超级用户权限

权限体系有一个基本的例外，即超级用户 root，超级用户就是 uid=0 的用户。一般情况下这个用户名字被称为root，不过实际上也可以是任何其他名字，只要设置其uid为0。作为系统的管理者，root 可以访问任何文件并对其读写。而实际中讨论的攻破一个UNIX/Linux 系统，最主要的任务就是获得 root 权限，比如拿到 root 密码或者获取一个具有 root 权限的 shell。

总的来说，这个权限体系是比较简单的，不像 Netware 之类的文件服务器操作系统那样做到了精确的用户个人访问控制，但在实际应用中也问题不大。不过，UNIX/Linux 操作系统实际设计的时候，为了完成某些操作，增加了几个特殊的功能，其中最重要的是setuid和setgid，这是用户权限体系中主要的问题。

setuid、setgid和用户进程的权限有关。如上所述，每个文件有定义好的用户存取权限，但是用户只能通过程序存取相应文件，因此权限体系实际上是和用户进程打交道。UNIX/Linux 为每个用户进程分配一个用户 ID 和一个组 ID，进程需要访问文件的时候，就按照这个用户 ID 和组 ID 来使用权限功能。正常情况下，这个用户 ID 和组 ID 会被分配成执行对应命令的用户的uid和gid，从而维持权限体系的正常运转。

问题是有些命令必须能够绕过正常的权限体系才能执行。例如，passwd 命令可以让用户修改自己的密码。但是密码密文是存放在/etc/shadow 文件中，这个文件只允许超级用户读写，这样passwd命令必须无论谁来执行都自动具有root的uid。为了解决这类问题，UNIX使用了setuid（suid）机制。

3. suid 机制

suid 机制就是在权限组中增加 suid/sgid 位，凡是 suid 位被置 1 的文件，当它被执行的时候，自动获得文件属主的uid；同样，sgid 被置位，也能自动获得文件属组的gid。不过实际上用sgid 的很少，主要还是用suid。

由于suid的这个特性，在实用中很容易带来安全性问题，尤其是如果setuid程序被溢出（见下节），或者利用环境重定义技术破解就很可能导致被非法用户得到一个具有root uid的进程。当这个进程是一个终端shell的时候，用户就得到了系统的完全控制权，即系统被攻破了。相应地，如果攻击者侵入了系统，也许会留下一个具有root suid 的 shell 程序，作为以后控制系统的入口（虽然这个办法现在不那么常用了）。

为了安全，应尽量少用 suid 功能，并且定期检查（可以使用专门工具）系统上有没有来源不明的suid程序，特别是绝对不要写一个suid的shell脚本，这是一种灾难性的行为。

超级用户和 suid/sgid 机制的存在，是 UNIX/Linux 体系设计上的一个弱点，这可以用一些手段加以弥补，例如后面提到的 LIDS 系统。

5.1.4　存储空间安全

这里说的存储空间安全，指的是内存及外存储设备的安全，包括越过文件系统直接访问裸设备的保护。其中，最重要的是内存安全性。

1．内存安全性

内存安全性可以分成两个部分，一个是禁止内存中的重要数据被窥探到，另一个是保证进程不去执行非法指令。特别是后者，是多种攻击与保护技术的核心内容。

由于 Linux 是个多用户多任务操作系统，每个进程都会享有自己的地址空间，一个进程不能访问另一个进程的地址空间，而属于操作系统的地址空间是受保护的，所以原则上，每个进程只能看到属于自己的和公用的内存数据。

但是任何功能实行起来都会有例外。每个进程都可能要分配动态内存，并且在申请到的内存中写入自己的数据。当进程退出的时候，这些内存被释放，但是内存中仍然会留着进程的临时数据。只要这些内存不是马上被下一个进程使用，下一个进程是看不见这些数据的。因此为了避免出现影子，操作系统可以设计为每当给进程分配内存的时候，都首先将分配的内存清 0，释放的时候也清 0。这样，似乎就堵死了用动态内存泄露数据的可能性。

问题在于，为了调试的方便，UNIX 系统使用了一个名叫 core dumped 的功能。简单地说，当发生像访问空指针、指针越界、试图写入保护数据之类进程访问不属于自己的段地址的数据时，UNIX 会直接中止这个程序。为了让程序员可以分析和调试定位这种错误，操作系统会把事故现场（出错时分配给进程的内存数据）复制到一个名叫 core 的文件中，保存在当前路径下，并且返回一个 core dumped 信息。这样，如果事故现场中包含了一些敏感的信息，就可能导致灾难。例如，最早的 ftpd 程序就存在这类问题，使得用户可以从 core 文件中获得 shadow 文件的内容。

Linux 继承了这种 core dumped 机制，不过为了保险，管理员可以用 ulimit 命令选择限制 core 文件的大小。

命令格式如下。

```
ulimit -c [size]
```

如果限制为 0，就禁止了 core 文件的产生，例如：

```
ulimit -c 0
```

当然，普通用户也可以限制自己的进程不产生 core 文件，用如下命令：

```
limit coredumpsize 0
```

现在来考虑内存安全的另一个也是更严重的问题——缓冲区溢出的问题。简单地说，缓冲区溢出就是利用程序的漏洞，用某些恶意代码覆盖程序代码，从而攻击系统。称为缓

冲区溢出，主要原因是这种技术都是利用向缓冲区中写入过量数据的方式完成的。这个技术的原理可以通过下面的例子来理解。

通过往程序的缓冲区写超出其长度的内容，造成缓冲区的溢出，从而破坏程序的堆栈，使程序转而执行其他指令，以达到攻击的目的。造成缓冲区溢出的原因是程序中没有仔细检查用户输入的参数。例如下面的程序代码：

```
void function(char *str) {
char buffer[20];
strcpy(buffer,str);
}
```

注意到代码中 buffer 是个定长数组，而 str 的长度是未知的。strcpy 直接把 str 中的内容复制到 buffer 中，由于 C 语言编译器并不会进行数组边界检查，如果 str 的实际长度大于 20，那么 strcpy 复制了 20 个字符后，并不会就此停下来，而是继续复制直到 str 被全部复制。当然，除了前 20 个字符在 buffer 中外，其他的内容会继续存入后续内存地址，从而修改了其他部分的数据/代码。如果在 buffer 之后是一段执行代码，则程序就被修改了。如 strcat、sprintf、vsprintf、gets 和 scanf 等涉及字符串传送的函数，都会有这个问题。

正常情况下，这种缓冲区溢出只会造成一个 segmentation fault，然后是 core dumped。不过，精心设计的溢出程序会在缓冲区溢出后让系统执行一些特定的代码，特别是会产生一个 shell，这样，如果被溢出的程序是个 root 权限的程序的话，那么系统就被攻破了。

2. 基本的溢出攻击方式

基本的溢出攻击方式按照其使用的程序空间类型可以分为如下三种。

1）堆栈溢出

堆栈溢出是最重要的缓冲区溢出攻击手段，它的思路是利用堆栈返回地址。熟悉汇编语言的读者会知道，堆栈是一段内存空间，一个程序执行的时候，内存可以大概地分成代码、数据段和堆栈段。

代码段（内存低端）
数据段
堆栈段（内存高端）

数据段存放着程序的静态数据，而动态数据则从堆栈中分配，分配过程是从低端到高端。另一方面，函数使用的参数和返回地址则用堆栈高端存放。当发出一个函数调用的时候，计算机做如下操作：首先把参数压入堆栈；然后保存指令寄存器（IP）中的内容作为返回地址（RET）；第三个放入堆栈的是基址寄存器（FP）；然后把当前的栈指针（SP）复制到 FP，作为新的基地址；最后为局部变量留出一定空间，把 SP 减去适当的数值。而函数返回的时候，简单地释放局部变量，然后从堆栈中取出返回地址，用 ret 命令回到调用点。这样，因为缓冲区是在堆栈底部，溢出程序只要溢出足够多的数据，就修改位于内存高端的堆栈返回地址，从而，当函数返回的时候，就会跑到溢出者设置的返回点。假如那里放着执行有 shell 命令的代码（因为这个地址是由溢出者设计的，所以可以选择让它指向某个缓冲区，其中存放了溢出者准备的代码），攻击者就拿到了一个 shell。如果这个进程是个

root 进程，攻击就成功了。

2）函数指针

C 语言可以定义函数指针，执行某个地址上的任何函数。但是这个指针可以指向任何地址空间，所以攻击者可以用缓冲区溢出的办法改写指针内容，指向自己准备的 shell 代码。当程序执行到调用函数指针部分的时候，攻击者就取得了控制权。

3）长跳转点

setjmp/longjmp 是一个 C 语言的非结构化跳转系统，允许程序员设置一个跳转点，然后可以从任何函数内部直接跳入这个跳转点。这个功能的存在主要是为了处理程序事故(例如处理紧急信号)。然而，跳转点可以指向任何地址，如果 setjmp 设置的跳转点被攻击者修改，当激活 longjmp 函数的时候，就会跳向攻击者设置的地址。

3. 溢出攻击解决方案

从理论上说，让数据段不可执行就可以避免绝大部分缓冲区溢出的发生，然而这是不现实的，因为现代的编译器为了效率，不可避免地要把部分可执行代码动态地放入数据段中，简单的禁止数据段执行就会导致这样的程序无法运行，所以只能考虑一些折中的办法。

1）补丁

最常用的操作系统补丁是设置堆栈段为不可执行，由于没有多少程序会把代码放入堆栈中，这样的方法基本上是安全的。但是有一个重要的例外，即在 Linux 中，向进程发送信号的代码是被放入堆栈中的，这是操作系统设计的问题。对应的堆栈禁止执行的补丁被设置为发送信号时暂时允许堆栈执行，幸好这一般不会带来重要的安全问题。这个补丁是 Sun 开发的，不过它并没有被内核开发组直接使用，需要自己下载这个补丁并重新编译内核。另外，有报道说 gcc 的某些功能在这个补丁作用下将会失效。

遗憾的是，这个补丁对于不使用局部自动变量的溢出攻击没有什么效果，攻击者可以将代码置入堆或者静态数据中，对这类攻击，还需要其他的应付措施。

2）编写程序

对于一般意义的溢出攻击，事实上只有一个可靠的办法，就是编写的程序不会溢出，例如在 copy 字符串之前首先检查字符串的长度，运行期动态检查数组边界等。不过，即使你做了这些，仍然不能保证系统库函数中不存在可能被溢出的代码，特别是 libc 和 glibc。一旦这些库被攻破，后果通常是灾难性的（例如，不久以前的 printf 格式化字符串漏洞）。幸好，一般当发现这些漏洞之后，很快就会有对应的补丁程序发行。

总之，缓冲区溢出是所有安全问题中最重要也最危险的攻击手段，几乎所有“致命的”安全漏洞都和它有关。相应地，许多程序因为有容易被溢出的名声，被安全性分析看成是麻烦的根源，这些程序如下。

- bind：DNS 服务程序，这是最著名的易遭攻击的程序。
- wu-ftpd：ftp 服务程序。
- rpc 服务程序：Sun RPC 服务程序，如 rpc.statd 等。

另外，诸如 apache、sendmail 等程序因为过于复杂，也容易出现缓冲区溢出漏洞。

3）提升安全级

解决缓冲区溢出的另一个方案，是前面讲的解决超级用户弱点的办法，即通过把系统

安全性提升到B级，使得即使攻击者溢出成功，也拿不到 shell 或者不能执行任何命令，从而减少被攻破的可能性。具体的细节实现在5.1.6节中介绍。

4）利用文件系统和磁盘分区

另外一些安全性问题与文件系统及磁盘有关。Linux 定义了“单用户”模式，使得用户不输入超级用户口令就可以直接以单用户方式启动系统。当然，解决方法是在 lilo 中使用 restrict 选项和 password。不过，由于 ext2 文件系统是自由的，总还存在用软盘启动系统并直接用命令 mount 根文件系统的可能，这个问题是没有解决办法的，毕竟，没有物理安全性就没有一切。

UNIX/Linux 系统有个著名的问题——链接。链接就是文件可以有多个名字。这本身没有什么问题，不过在 UNIX/Linux 系统中，硬链接是可以越过原来的文件名直接存取文件的，所以考虑下面这种情况：某个超级用户进程使用 file1 作为临时存取文件，然后某个用户把某个重要文件链接到了这个名字下面，结果，当超级用户进程运行的时候，系统就被破坏了。这最容易发生在/tmp 目录下，因为系统管理进程会不断地写这个目录。

解决的方法是将普通用户可写的目录/文件放到独立的分区中。因为硬链接是不能跨越文件系统边界的，这样就可以避免上述的问题发生，通常/tmp、/var 和/home 等都会被做成独立的文件系统。

5）chroot

最后介绍一个解决文件安全的“最终”解决方案：chroot。这个技术可以让用户程序把某个目录当成根目录，从而无论怎样切换也不能越出这个虚拟的根目录，也就不会危害其他目录下面的文件。这个方法是可靠的，比如 ftp 程序在处理匿名用户访问的时候就会使用这种办法把匿名用户锁在 ftp 目录中。遗憾的是，这种办法因为用户不能越出虚拟根，也就不能访问其他目录下的文件，从而对任何需要访问的文件都必须把它复制到这个虚拟根下面，使得存取变得非常烦琐，因此只用在少数场合。

5.1.5 数据的加密

为什么要对数据进行加密?部分原因是交换数据的需要。如果要通过某种不可信的方式（例如网络、磁带等）交换数据，就必须保证不能让一般人随便窥探数据。这种情况下，基于权限的保护方式已经不能使用，只能通过加密来提供额外的保护。

首先要说明的是，在一般情况下，加密这个词汇除了表示将信息无损地变换为密文之外，还包括一些所谓单向算法，后者其实就是一个计算校验和的过程。换句话说，知道单向算法的结果，是没有可能逆向求出原文的（所以这个算法被称为单向的）。几乎所有的单向算法都使用了与 HASH 函数相关的技术，这种技术使得事实上完全不可能有两组不同数据产生同样的校验和。另一方面，除了实际计算一下以外，也不可能有别的办法生成某段数据的校验和。因此，这种技术被用于确保数据的可信性，或者说，确信数据在传输过程中未经修改。

1．MD5

最常用的单向算法是 MD5，相应的程序实现在 Linux 中就是 md5sum 校验程序，它可

以为一个文件生成 md5 校验和。

```
[root@ cs bin]# md5sum ~wly/wly.rar
4fbec4d2dedcc61efec6e6aadf595e96 /home/wly/wly.rar
```

输出的 128 位十六进制数就是相应文件的 md5 校验和。

非单向（可逆）的加密算法也可以分成两类。一类称为对称算法，它在加密和解密的时候用的是同样的加密密钥，就是对密文用同样的密钥变换之后可以得到原文。另一种是非对称的，就是要使用两个密钥，用一个密钥加密的数据只能用另一个密钥解密，反之亦然。当用户需要给别人发送信息的时候，将某一个密钥发送给别人，然后用另一个密钥加密文件，接收方只要用收到的密钥解密这个文件就可以了。

两种加密手段都有其代表技术，如 DES 是对称算法，RSA 则是非对称算法。现在许多加密工具都是基于这两种技术的。不过应该指出的是，由于法律上的一些问题，DES 加密系统使用的并不是很广（主要因为美国出口限制），而且大部分 DES/RSA 加密工具是硬件实现的。除此之外，DES 加密系统的可靠性仍然在争论中。因此，实际应用中，特别是类似电子邮件之类的数据交换，大家使用的主要是一种名叫 PGP 的加密工具，这个工具是由 Phil Zimmermann 发明的（他由于发明这个工具一度遭到诉讼）。

2. PGP

让我们回过头来考虑双密钥体系，标准的工作方式是：用户 A 首先要建立两个密钥，一个称为私用密钥，由用户保留；另一个称为公用密钥，对外公开。当用户 B 想要向用户 A 发送信息的时候，他用 A 的公用密钥来加密数据，然后传送给 A。这样，由于只有 A 具有私用密钥，所以其他人（包括 B）无法解密对应密文，这就保证了传送的数据只有 A 可以看到。不过，由于公用密钥是公开的，任何人都可以冒充 B 来给 A 发信，所以这个体系还存在可能被假冒发送信息的弱点。解决的方法很简单，就是利用前面说的单向算法给数据加上签名。这就是 PGP 的基本思路。

PGP 的算法是这样的：首先，B 和 A 各生成一对密钥，然后通过一个可信的机构，交换自己的公用密钥（当然，两人也可以直接见面解决这个问题），在这之后，如果 B 想要向 A 发送自己的信息，那么，他首先要用 MD5 算法计算出整个信息（严格地说，在 PGP 中，计算的是信息+发送者的 ID+日期时间等）的校验和，然后用自己的私用密钥加密这个校验和并且附加在整个信息后面；接着，再用 A 的公开密钥加密得到的全部数据，最后发送给 A。而当 A 接收到这个数据的时候，他首先用自己的私用密钥解开密文，得到原文和加密后的校验和；然后再用 B 的公钥解密校验和；最后重新计算校验和，判断信息是否被修改过。

这个算法本身可以说无懈可击，除了交换公钥问题（这可以通过建立权威的交换中心的办法解决）。但是实际上，不对称的双密钥体系的加密解密速度一般是很慢的（特别是 RSA），因此 PGP 采用一个折中的办法，对信息加密采用一个名叫 IDEAL 的算法。IDEAL 是一个传统的单密钥对称算法，其计算速度很快。实际计算中，PGP 首先产生一个随机密码；然后用这个密码作为密钥对信息进行 IDEAL 加密；接下来，用 RSA 算法对密码进行加密。解密的时候，接收方是先用 RSA 算法得出随机密码，然后用 IDEAL 算法对信息进

行解密得出源信息。

目前，PGP 源码可以从因特网上自由下载，也可以使用编译好的 PGP 包。

1）利用 PGP 程序的源代码或可执行文件数据进行加密

第一个任务是生成公用密钥和私用密钥，可以使用 pgp -kg 命令生成。

```
#mkdir.pgp
#pgp -kg
```

（1）确定 PGP 密钥的长度，可以选择 512、768 或者 1024 位，一般都选择 1024 位，因为 PGP 加密的速度相当快，所以选择 1024 位不会造成性能上的损失，显示如下：

```
Pick your RSA key size:
     ① 512 bits-Low commercial grade, fast but less secure
     ② 768 bits-High commercial grade, medium speed, good security
     ③ 1024 bits- "Military" grade, slow, highest security
Choose1, 2, or 3, or enter desired number of bits:3
```

（2）需要设置用户名和口令，用户名可以自己随便设置，口令是用来保护私用密钥的，原则是尽量难猜一点，而且一定要记住自己的用户名和口令，显示如下：

```
Enter pass phrase:
Enter same pass phrase again:
Note that key generation is a lengthy process.
```

（3）PGP 会要求输入随机数字密钥，因为没有人喜欢输入这样长的数字，PGP 的办法是让你随机地按键盘上的键产生出一个数字串，所以可以任意按键直到 PGP 的提示数字变成 0，显示如下：

```
We need to generate 1215 random bits. This is done by measuring the
time intervals between your keystrokes. Please enter some random text
on your keyboard until you hear the beep:
1215
```

然后，就会在用户的 PGP 目录下面产生出私用密钥的存储文件和公用密钥的存储文件，名字分别是 pubring.pgp 和 secring.pgp。

2）把某个文件加密传送给别人

（1）首先要生成公用密钥输出文件。

```
Pgp -kx wly public
(wly是我们设定的pqp用户名)
```

这样就生成了一个用来传送公用密钥的文件 public.pgp。

（2）用 PGP 来加密数据。例如，要和 user1 通信，首先要把自己的 public.pgp 交给 user1，然后 user1 需要把这个公用密钥加入他自己的密钥环。

```
 pqp -ka public.pqp
```

PGP 会产生烦琐的询问，主要是问你是否愿意相信这个密钥，选择同意之后，这个密钥就被加入到 user1 的密钥环中。

（3）user1 就可以用 PGP 加密一份文件了。例如，想把 mychains 文件加密后交给 toot，需要用 root 的公用密钥对 mychains 进行加密。

```
pgp -e mychains wly
```

这个过程会产生一个 mychains.pgp 文件，然后 user1 可以把这个文件发送给 root。

（4）root 接收到这个文件之后，可以对其进行解密。

```
pgp mychains.pgp
```

这个动作将产生原来的 mychains 文件并且删除 mychains.pgp 文件。

3）PGP 的其他参数

PGP 有很多参数，比较重要的介绍如下。

- pgp -e [a] 源文件 收件人 ID [其他收件人 ID]：用收件人的公用密钥加密源文件，得到密文。这个过程将会生成一个源文件名加上.pgp 后缀的二进制加密文件。如果想把加密的结果通过电子邮件传送给别人，可以加上 a 参数。
- pgp -ea 文件名字 收件人 ID：然后将得到 7 位编码的密文，后缀也变成了.asc。
- pgp -s [a] 源文件 [-u 我的 ID]：用我的私用密钥给源文件签名，如果不用默认用户名，就要用-u 参数指定某个特定 ID 的私钥。同样，加上 a 参数可以得到 7 位编码输出。
- pgp -se [a] 源文件 收件人 ID [其他收件人 ID] [-u 我的 ID]：即先签名再加密，参数意义同上。
- pgp -sb [a] [+clearsig=on] 主文件 [-u 我的 ID]：产生与主文件分开的签名文件，a 与-u 参数的意义同上。+clearsig=on 参数的意义是当为一份二进制主文件签名时，生成的签名是 7 位可识别的形式。这个参数也可以在 pgp.ini 文件中设置。
- pgp -c 源文件：用 IDEA 加密算法对源文件加密，由用户给出口令。这里没有使用 RSA 算法，因此只是一种传统加密。PGP 也提供了这样一种相当不错的加密手段。
- pgp -a 源文件：用 RADIX 64 编码对原文件编码，输出文件默认用.asc 作为扩展名。RADIX 64 编码和 E-mail 中常用的 MIME BASE64 编码是兼容的，用 MIME 的解码工具可以解开 PGP 的编码。
- pgp [-d] [-p] [-b] 密文：解密或检查签名，-d 参数用来保留密文（默认为删除密文），-p 参数用来恢复源文件加密时的文件名。
- pgp -b 被签名后的文件：从被签名的文件中分离出签名文件。
- pgp -kg [密钥长度]：生成密钥对，可以给出长度。
- pgp -ka 密钥传递文件：把别人给你的密钥传递文件中的密钥加入到你的 pgp 密钥环。
- pgp -kx [a] 用户名 密钥传递文件：生成自己的密钥传递文件，以后只要把这个文件交给别人，对方就可以用你的公用密钥对文件进行加密。加 a 参数将产生一个用 7 位编码的密钥文件，从而方便通过 E-mail 传递它。

- pgp -kv [v] [用户名]：查看密钥列表。v 参数代表详细列出附着的签名。
- pgp -kvc [用户名]：列出密钥的“指纹”。
- pgp -kc [用户名]：列出密钥环的内容和检查签名情况，例如某人的信任参数等。
- pgp -ke 用户名：编辑密钥环中密钥的用户名或者更换口令。
- pgp -kr 用户名：删除一个用户。
- pgp -ks 对方被签名 ID [-u 你用来签名的 ID]：对一个公用密钥进行签名认证。
- pgp -krs 用户名 [密钥环]：删除一个签名。
- pgp -kd 你的 ID：永久性地废除一个密钥，并且生成一个“废除证明”，用生成文件通知外界。

5.1.6 B1 安全级强化

本节介绍一个对 Linux 安全问题的解决思路。对于 UNIX/Linux，攻击和防御的要点都是围绕着 root 用户及如何获得一个 root shell 展开的。这里的关键是 root 可以做任何事情，而所有的重要操作又都需要 root 权限，这就使得一旦侵入者拿到 root shell，系统就被完全攻破了。

因此可以考虑这样的技术，将系统的敏感任务分别划给不同的用户，取消全能的 root 账户的存在，避免不同的安全问题纠缠在一起。这样即使攻击者进入系统，也无法危害系统的安全性。实现这种功能的最简单方法是将系统安全性提升到 B 级，这样可以在很大程度上减少系统中的安全隐患。

LIDS 介绍

实现这种功能，需要修改 Linux 的系统内核，目前，这种修改都是以补丁的方式发行的，这里介绍的是谢华刚的 LIDS（Linux 入侵检测系统）。应该指出的是，LIDS 更多的并不是一个“入侵监测系统”，而是一个安全性补丁，它可以实现下列的功能。

- 入侵防护：可以对系统中重要的文件、进程和设备进行保护，即使是入侵者已经拿到了 root 账号，也能保证这些对象不会被入侵者操纵。
- 入侵检测：LIDS 可以提供对于端口扫描的检测。
- 入侵响应：当发现系统受到攻击之后，它可以将必要的信息记录到日志文件中。

对于我们来说，入侵防护是最重要的功能。这是通过修改系统核心调用函数 open、mknod 和 unlink 等的代码来实现的。一旦 LIDS 开始工作，它在内核中维护一张授权表格，当有某个程序调用 open 来打开文件的时候，LIDS 将会检查授权表，阻挡一切未经授权的访问。由于这个动作发生在内核中，即使超级用户也无法绕过授权工作，除非将 LIDS 关掉。同样，在访问进程和设备的时候，也发生授权检查过程。

授权规则由主体（subject）、客体（object）和授权构成，主体是发起操作的程序，客体是它准备操作的目标，而授权则是由 DENY/READ/APPEND/WRITE/IGNORE 几个级别构成的，比如 READ 是允许主体读这个文件，WRITE 允许写，而 IGNORE 则忽略存取规则。授权级别之间的关系是后面的包含前面的，也就是说允许 WRITE 也就直接允许了 READ 和 APPEND。当然，DENY 就是没有任何授权。

LIDS 的文件授权规则是基于节点号的，也就是实际控制的不是文件名而是 i-node，因此用户不能通过建立链接或者改名之类的办法绕过存取授权。当然，如果文件发生了移动，也只有刷新授权列表才能保证 LIDS 正确工作。

除此之外，另一个重要的任务是控制用户程序的"能力"的功能。所谓"能力(capability)"，是指程序控制系统资源的权利，如访问网络，直接存取设备等。表 5-6 是已经实现的能力的一览表。

表 5-6　已实现能力一览表

能　力	描　述
CAP_CHOWN	允许无限制使用 chown 改变文件所有权
CAP_DAC_OVERRIDE	允许无条件的文件访问（无 DAC 限制）
CAP_DAC_READ_SEARCH	允许所有相关读/搜索动作，忽略文件许可
CAP_FOWNER	即使属主 id!=userid 也允许文件访问
CAP_FSETID	允许对任何文件设置 setuid/setgid 标志
CAP_KILL	允许向非自己所拥有的进程发送信号
CAP_SETGID	允许无限制的 setgid（2）和 setgroups（2）
CAP_SETUID	允许无限制的 setuid（2）和 friends
CAP_SETPCAP	允许将你拥有的任何能力传递给另一个 PID
CAP_LINUX_IMMUTABLE	允许修改不变的和附加的文件属性
CAP_NET_BIND_SERVICE	允许绑定小于 1024 的 TCP 和 UDP 端口
CAP_NET_BROADCAST	允许发往外部的广播包
CAP_NET_ADMIN	允许与网络接口有关的许多选项，例如路由表修改等
CAP_NET_RAW	允许使用原始套接字和数据包套接字，例如手工构造的数据包
CAP_IPC_LOCK	允许锁定共享内存段
CAP_IPC_OWNER	允许无限制 IPC 访问
CAP_SYS_MODULE	允许插入、移除 LKMs
CAP_SYS_RAW IO	允许直接访问设备，例如/dev/［hs］da*
CAP_SYS_CHROOT	允许使用 chroot（2）
CAP_SYS_PTRACE	允许使用进程追踪任何进程
CAP_SYS_PACCT	允许配置进程记账系统
CAP_SYS_ADMIN	允许许多限制性的活动，例如设定 hostname，使用 mount，创建设备等（类似于 root 能力）
CAP_SYS_BOOT	允许使用 reboot（2）
CAP_SYS_NICE	允许提高优先权，影响非属主进程的 nice level
CAP_SYS_RESOURCE	通过 resource/quota/etc 限制禁止访问
CAP_SYS_TIME	允许时钟处理
CAP_SYS_TTY_CONFIG	允许 tty 设备配置
CAP_HIDDEN	LIDS 特定的能力，用于隐藏/proc（或者其他）中的一个进程
CAP_INIT_KILL	LIDS 特定的能力，用于限制向 init 拥有的进程发送信号，通常为 daemons

能力的设计可以使管理员把原来属于同一个超级用户 root 的功能分配给多个用户（准确地说是多个进程），彼此互不干扰，避免系统某一点被攻破就全盘崩溃。特别是许多任务可以交给非 root 用户，从而避免了由于 bind 之类的程序被攻破导致系统被全面攻破。

由于在实际操作中，许多程序要装入其他的程序来实现功能，因此 LIDS 中定义了两个特定的权力：INHERIT 和 NO_INHERIT，它们决定当前进程的授权规则和能力是否传递给子进程。

LIDS 对系统的另外一个保护功能是封装内核，简单地说就是禁止装入内核模块的能力。许多 Linux 黑客程序是采用修改内核的方式（加载内核模块）获得系统控制权的，这个功能可以用来避免这类程序的危害。

LIDS 详细的资料和安装/配置说明可以在 http://www.lids.org 找到，其中还有一些对于不同的服务器系统如何配置 LIDS 的样板。

5.1.7 日志

日志就是对系统行为的记录。例如记录某个用户的登录/退出及执行的命令，系统中发生的错误等。在标准的 Linux 系统中，操作系统维护三个基本的日志。

- 连接时间日志：用来记录用户的登录信息，这是最基本的日志系统，管理员可以利用它来跟踪谁在什么时候进入了系统。
- 进程记账日志：用来记录系统中执行的进程信息。例如某个进程消耗了多少 CPU 时间等。
- syslog：syslog 系统日志并不由系统内核维护，而是由 syslogd 或者其他一些相关程序完成。它是各种程序（主要是 daemon 进程）对运行中发生的事件的处理代码。

1. 连接时间日志

连接时间日志由 utmp、wtmp 和 lastlog 记录构成。有关当前登录用户的信息记录在 utmp 中，用户的登录和退出记录及系统开关机的记录存放在 wtmp 中，用户最后一次登录的信息存放在 lastlog 文件中。遗憾的是这些文件没有固定的存放地点，尽管在大多数 Linux 系统中，它们放在/var/log 下面，但是这并不是一个标准。

这几个文件都是二进制文件，所以不可能用简单的 vi 和 cat 命令查看它们的内容。相反，有几个标准命令可以完成这些功能。

- w 命令：显示当前用户及其登录终端/进程信息。

```
[wly@ cs log]$ w
11:25pm  up 2 min,  1 user,   load average:0.50, 0.24, 0.09
USER TTY      FROM              LOGIN@     IDLE    JCPU   PCPU    WHAT
wly   pts/0   202.206.196.226 11:25pm     1.00s   0.26s  0.05s    w
```

- who 命令：显示当前用户及其登录终端。

```
[wly@ cs log]$who
wly      pts/0   Oct 28 23:25(202.206.196.226)
```

- users 命令：显示当前用户列表。

```
[wly@ cs log]$ users
wly
```

- last 命令：显示自从 wtmp 命令建立（通常是系统启动）之后登录的用户信息。

```
[wly@ cs log]$ last |more
```

```
wly     pts/0       202.206.196.226    Mon Oct 28 23:25        still logged in
wly     tty1                           Mon Oct 28 05:56 -down  (00:01)
wly     pts/0       202.112.94.108     Wed Oct  2 12:18 -14:33 (02:15)
wly     pts/1       202.112.94.108     Wed Oct  2 11:52 -12:03 (00:10)
vista   ftpd23238   202.206.196.221    Wed Oct  2 11:03 -11:07 (00:04)
vista   ftpd23235   202.206.196.221    Wed Oct  2 10:50 -12:02 (01:11)
vista   ftpd23234   202.206.196.221    Wed Oct  2 10:18 -11:09 (00:50)
vista   ftpd23233   202.206.196.221    Wed Oct  2 10:17 -11:17 (01:00)
vista   ftpd23232   202.206.196.221    Wed Oct  2 10:17 -11:04 (00:47)
vista   ftpd23231   202.206.196.221    Wed Oct  2 10:17 -11:05 (00:48)
vista   ftpd23229   202.206.196.221    Wed Oct  2 10:17 -10:17 (00:00)
vista   ftpd23230   202.206.196.221    Wed Oct  2 10:16 -11:08 (00:51)
vista   ftpd23228   202.206.196.221    Wed Oct  2 10:16 -11:05 (00:48)
vista   ftpd23227   202.206.196.221    Wed Oct  2 10:16 -10:50 (00:34)
wly     pts/0       202.112.94.108     Wed Oct  2 10:03 -12:10 (02:06)
ftp     ftpd22930   linux.asnc.edu.cn  Wed Oct  2 04:03 -04:03 (00:00)
wly     pts/0       202.112.94.108     Tue Oct  1 20:04 -20:17 (00:12)
wly     pts/0       202.112.94.108     Tue Oct  1 17:33 -19:37 (02:03)
wtmp begins Tue Oct 1 08:50:27 2002
```

- ac 命令：显示所有用户总的联机时间，如果加上-p 参数，则显示每个用户的总联机时间。

```
[wly@ cs log] $ ac
total       3966.60
[wly@ cs log] $ac -p
 why                0.77
 zhp                0.09
 ftp             3906.91
 wly                0.25
 root              58.45
 apache             0.01
 jly                0.13
 total           3966.61
```

- lastlog 命令：显示所有用户上次登录的时间，如果用-u 参数，则显示单独的某个用户的登录时间。

```
[wly@ cs log] $ lastlog -u wly
Username        Port     From              Latest
wly             pts/0    202.206.196.226   Mon Oct 28 23:25:29 +0800 2002
```

2. 进程记账日志

进程记账功能来源于 BSD 系统，这个功能必须在内核中被激活，这是靠编译内核中启

用 BSD Process Accounting 完成的，图 5-1 所示是内核编译设置图。



图 5-1

重新编译内核之后，还要用命令启动记账。

```
#/sbin/accton [记录文件]
```

例如，#/sbin/accton/var/log/pacct 将打开进程记账系统，记账信息存放在/var/log/pacct 文件中。注意，在启动记账之前，/var/log/pacct 文件必须存在，可以用 touch 命令自己建立它。

```
#touch/var/log/pacct
```

不带参数的 accton 程序关闭进程记账功能：

```
#/sbin/accton
```

设置了记账之后，可以用 sa 命令显示系统执行的进程和每个进程消耗的时间，但是对于我们来说，更有用的是 lastcomm 命令，它可以显示以前执行的命令。

3. syslog 日志

最后一种也是最有用的日志工具是 syslog 相关日志。当然，这种日志并不全都通过 syslogd 进程。例如，httpd 程序就会绕过 syslogd，自己去写 log 文件。不过，这种日志工具都是对程序运行时发生时间的一种记录，而且完全是文本文件。

通过函数 syslog 调用发生的日志记录都会通过 syslogd 程序写入系统日志，具体的写入方式由/etc/syslog.conf 控制，这个文件由一系列行组成定义哪些消息被写到哪里，以#开头的行被当成注释行。

正文行的语法如下。

[消息类型]　　[动作]

中间用 Tab 字符分割。

消息类型由“消息来源”和“优先级”构成，中间用一个点号连接。如果想在一行上标出多个消息类型，中间用分号分开。

消息来源如表 5-7 所示。

表 5-7　消息来源一览表

来　源	描　述
kern	内核
user	用户程序
mail	电子邮件系统（smtp、pop3 等）
daemon	系统守护进程
auth	和安全性及权限修改相关的命令
lpr	打印机
authpriv	私用的授权信息
mark	定时产生的时间戳
cron	cron 程序
syslog	syslod 程序自身产生的消息
local0-7	8 种本地消息
news	Usenet 系统消息
uucp	uucp 程序

可以使用一个通配符来指代所有的设备，例如“*”代表一切消息来源，而 none 表示什么都没有。

优先级被分成下面的几个等级，匹配规则是向上包含，也就是说默认条件下定义低优先级事件将同时包含高优先级事件。除非定义了“=”操作符，例如 kern.=info，将只匹配 info 级别的消息，而 kern.info 将同时匹配所有 info 以上级别的 kernel 消息。优先级别如表 5-8 所示。

表 5-8　优先级表

级　别	描　述
emerg	最高优先级别
alert	紧急状态
crit	资源临界
err	出现错误
warning	警告
notice	出现了某些不寻常的事情，可能应该调查
info	一般性消息
debug	用于调试的信息

“*”和 none 仍然可以使用。

“动作”选项告诉 syslogd 应该如何处理对应的消息：将它存入硬盘文件，显示在终端

上，发送给某个用户或者转发给另外一台主机。表5-9是可用动作的一览表。

表5-9 可用动作的一览表

选 项	描 述
文件名	写入某个文件，注意这里必须使用文件的绝对路径
@主机名	转发给另外一台主机上的syslogd程序，这需要远端主机使用syslog -r命令启动syslogd
@IP地址	等价于@主机名，只是使用IP地址标志远程机器
用户列表	用逗号分开的用户列表，例如user1、user2表示如果user1和user2登录的话就把信息发送到他们的终端上
*	发送到所有用户的终端上
/dev/console	发送到本地机器的屏幕上
|程序名	通过管道转发给某个程序

日志不能增加系统的安全性，而最多只能增加攻击者被抓住的可能性。遗憾的是，无论哪一种日志，都会记录大量无用的数据，使得许多人失去认真检查它的兴趣，解决的办法是使用某种专用日志检查工具，如swatch。

5.2 网络安全

5.2.1 网络接口层

1．监听

首先来考虑网络接口/物理层的主要问题。这两层完成的任务是将数据转化成物理的网络帧，并且在电气连接上传递。这两层的主要问题，在于具体的硬件特性。通常情况下，最主要的连接产品是以太网。

以太网的结构非常简单，许多网络接口卡（NIC）被电缆和集线器连接在一起。当某一台机器准备发送数据的时候，它首先根据电缆电平判断一下当前是否有其他网卡在工作，如果没有，就开始发送一段数据（称为以太网帧），然后暂停一下，再重复上述操作。如果有两台机器同时发送数据，会导致网络冲突，在这种情况下双方都会暂停一下（具体暂停的时间是个随机量），然后继续发送。

以前使用的同轴电缆（10-base5或者10-base2）的以太网，所有的网卡都被直接连接到电缆上。而现在的以太网（100-base TX及更先进的系统）中，网卡首先用双绞线连接到集线器和交换机，然后再连接到其他的机器。

集线器就是一个简单的信号放大器，用来补偿信号在传输中的衰减和变形。由于一台集线器可以连接很多的网卡（堆叠式集线器可以堆叠上百个插口），而集线器并不去辨认信号的出发点和目标，所以任何信号都会跑遍这个系统内所有的网卡和电缆（术语叫做“冲突域”）。

以太网卡使用一个48位的整数来标志自身，理论上它是唯一的，这个数字称为MAC

地址。当某一个网卡发送数据的时候，它在以太网帧中携带发送者和接收者的 MAC 地址。而接收方只是沉默地倾听，记录所有的以太网帧，并且试图分辨出接收者的 MAC 地址，如果这个地址和自身的地址相同，就会将这个网帧传送给上层网络驱动程序，否则，这个网帧将被简单地丢弃。

事实上处于倾听状态的所有网卡都能监听到任何同域内网卡发出的所有以太网帧，因此以太网中传输的任何数据实际都是公开的。一般情况下网卡会简单地丢弃不是发给自己的以太网帧，但是要知道这并不是在硬件上设置的，而仅仅是驱动程序的习惯。实际上，一切以太网卡都可以设置成不区分目标 MAC 而直接记录所有听到的以太网帧的工作方式，这样，如果你愿意，可以用程序得到所有本域内传输的数据。当然，这要求程序绕过 TCP/IP 堆栈和网卡直接打交道，这种工作方式叫做“混杂模式”。

上面说的在现代以太网中稍微有点改变。为了提高以太网的工作效率，用来连接网络双绞线的除了集线器之外还有交换机。交换机的能力比集线器强些，它除了补偿信号之外，还要辨认以太网帧的目标地址，并且送到正确的目的地。换句话说，以太网帧现在不能跑遍整个系统了。交换机还可以暂时存储以太网帧用来避开网络冲突。但是很显然，交换机最多只能管到自己的双绞线插口（交换机端口），所以只要把上面说的“域”换成交换机端口，其他的仍然成立。

这样，在一个以太网内，用一个正确设计的程序就可以监听到别人发送的信息，这种程序叫做“嗅探器（sniffer）”。现在有很多有效的嗅探工具，这些工具可以用来分析网络的运行，也可以用作专门的窃听工具。典型的有 tcpdump 和 sniffit 等。当然，这种程序做的事情还要更复杂一些，它们能够自己完成从以太网帧重组 TCP/IP 数据包的工作，如 sniffit 甚至还能自动过滤传送的文本信息，如密码等。

监听/嗅探的问题其实不仅限于以太网，因为 TCP/IP 是分层的，因此，设置在路由上的监听程序毫无疑问可以听到所有通过它的信息。这也是一种监听方式，虽然实际的例子不多。

2. SSL

容易看出，监听嗅探问题在以太网的层次上是没法解决的，只能从另一个角度去解决。事实上监听的主要目的是了解对方传输的信息内容，特别是对于 WWW、MAIL、FTP 和 TELNET 等协议，其中很多东西是简单文本传输。解决问题最基本的思路是将要传输的数据加密，然后再送入 TCP/IP，这样即使窃听者听到了所有的信息，它仍然无法解读这些信息到底是什么。

这种思路的基本框架和前面讨论的 PGP 系统很类似，首先要实现一个交换证书的体系（认证中心），用来证明访问的目标和自身的身份是可靠的（由于 IP 和 DNS 欺骗的存在，确保你访问的站点是你认为正在访问的站点是非常必要的）。目前这方面已经有了一个标准框架，称为 PKI（Public Key Infrastructure）。然后，传输的数据使用数据加密和数字签名，保证传输信息的可靠性和可信性。

由于大部分网络上数据的传输是通过 HTTP 协议完成的，而这个协议也最容易受到监

听，所以 Netscape 发明了一个称为 SSL（Secure Socket Layer，安全套接层协议）的协议来强化 HTTP 的安全性。它实际是基于 socket 的，安装在传输层和应用层之间，可以提供数据的加密传输。但是，它并不包含数据签名功能，后者需要浏览器另外实现。利用 SSL 进行 WWW 传输的协议有个专门的名字，叫做 HTTPS，SSL 一共有三个版本。

先来简单介绍一下 SSL 的工作原理。当一个客户试图和服务器对话的时候，它首先要执行一个握手过程：客户将一些配置信息（SSL 版本号、使用的加密算法及一些随机数据）发送给服务器，同样，服务器响应的时候要把自己的 SSL 版本号、加密算法、随机数据和证书（证明自己身份的信息）传送给客户，除此之外，服务器还要传送一个用自己的私用密钥传送的问候信息（SERVER-HELO）。

证书是什么呢？简单地说就是一组数据，里面至少存放两件东西：某个服务器的身份信息和它的公用密钥。这个证书必须通过其他的手段发送给用户——除非你直接相信某个服务器，那就是这个服务器自己给自己发证书了。一般的证书发送是通过一些著名的专业证书发送机构（认证中心）来完成的，认证中心用分级体系，有一些最高级别的认证中心叫做 root CA，它们负责给其他的认证中心发证书，然后下属的认证中心再给网络上其他的单位发证书。

客户得到证书之后，去认证中心取得服务器的公开密钥，然后解密 SERVER-HELO 信息。如果成功，说明对方的私用密钥正确，身份无误；否则，对方就是假冒的。

然后，客户需要用已经产生的随机数据产生一个信息，然后用服务器的公钥对它加密，结果送给服务器。服务器解密之后，就得到了对应的信息。双方利用这个信息进行协商，得出 master key。

最后，双方用 master key 产生出通话的密码，这可以通过任何算法完成，因为由于有变换算法存在，不需要相互通话。得到通话密码之后，以后的数据传输都可以用这个密码使用对称单密钥加密算法完成，双方只要传递密文就可以了。

总之，用非对称加密算法产生和交换密码，用对称算法传递信息。

SSL 因为是 Netscape 开发的，所以最早的实现也是 Netscape 完成的。Netscape 关于 SSL 服务的库称为 SSL-Toolkit。不过实际上大家用的更多的是一个名叫 openssl（也叫 ssleay）的库。这个库系统可以从 www.openssl.org 取得。

为了使用 SSL，服务器程序必须支持 SSL 和 HTTPS。最主要的页面服务器 Apache 包含两个支持 SSL 的附加计划，一个为 Apache-SSL，它集成了 Apache 服务器和 SSL；另一个为 Apache+mod_ssl，它是通过可动态加载的模块 mod_ssl 来支持 SSL。

3．SSH

SSH 是一个用来解决 TELNET/FTP 协议安全性的工具，它支持两种协议：SSH 和 SFTP，简单地说就是 telnet 和 FTP 的加密传输版本。

SSH 协议有两个版本：SSH1 和 SSH2，两者的功能基本是一样的，不过算法不兼容。需要明确地选择 SSH1 或者 SSH2 作为通信方式。无论哪一种，都提供两种级别的安全验证。

（1）基于口令的安全验证，也就是一个加密传输的 telnet。所有传输的数据都会被加密，只要给出用户名和口令，就可以连接到系统，其他方面也和 telnet 完全一样。

（2）基于密钥的安全验证。需要准备一对密钥，如果要连接到 SSH 服务器上，客户端软件就会向服务器发出请求，请求用密钥进行安全验证。服务器收到请求之后，先在该服务器的主目录下寻找你的公用密钥，然后把它和你发送过来的公用密钥进行比较。如果两个密钥一致，服务器就用公用密钥加密“质询（challenge）”并把它发送给客户端软件，客户端软件收到“质询”之后就可以用你的私人密钥解密再把它发送给服务器。用这种方式不需要在网络上传送口令。

SSH 保证了 telnet 命令的加密传输，除此之外，SSH 还支持 scp 命令。

scp 命令格式如下。

```
scp本地文件　远程账号@ 远程主机名
```

这个命令用来把本地机器上当前目录下的本地文件复制到远程主机对应账号的宿主目录中。

另外，SSH 的具体实现中，还包括一个加密的 FTP 协议，称为 SFTP。它的基本结构和 FTP 相同，但是所有传输数据都是加密的，而且允许远程运行服务器上的程序。

另外，作为一种安全工具，SSH 还可以在系统之间建立加密传输通道，从而为本来不安全的协议提供附加的安全性。

SSH 本身是个商业软件，不过也有对应的免费开放实现，可以从 www.openssh.com 下载，这是一个基于 openssl 的 SSH 软件实现。

4．VPN

监听并不一定发生在以太网中，整个 Internet 都是没有安全性保护的，你的信息可能在任何一个地方遭到窃听或者篡改。上面讨论的 SSL 和 SSH 都是在应用层解决问题的办法，这种办法尽管基本上是可靠的，但是却要求应用程序被完全重写以便支持 SSL 和 SSH。

TCP/IP 的分层结构提供了另外一种解决思路。事实上 TCP/IP 并不考虑信息的物理传输，实际上是个“虚”协议，可以在任何数据传输技术上运行，甚至可以将其他某种网络协议作为 TCP/IP 的物理传输信道，这就是所谓的“隧道”。当用另一个 TCP/IP 来实现隧道并且对其传输的数据进行加密的时候，就构成了虚拟专用网（VPN）的基本框架。

虚拟专用网被定义为通过一个公用网络（通常是因特网）建立一个临时的、安全的连接。它在因特网的一些机器（或者子网）之间建立一个虚拟连接，使得这些机器看上去就像是处在一个独立的网络中，其他系统无法加入。虚拟专用网可以帮助远程用户、公司分支机构、商业伙伴及供应商与公司的内部网建立可信的安全连接，并保证数据的安全传输。它能提供如下功能：

- 加密数据：保证通过公网传输的信息即使被他人截获也不会泄露。
- 信息认证和身份认证：保证信息的完整性、合法性，并能鉴别用户的身份。
- 提供访问控制：不同的用户有不同的访问权限。

目前，可以用来实现 VPN 技术的标准协议主要有 Microsoft 的 PPTP（点对点隧道协议），IETF 的 IPsec（Internet 安全协议）和 NEC 的 socks5。PPTP 基本上就是一个加密的 PPP 协

议，首先，IP 包被按照一般 PPP 拨入的方法封装进 PPP 包，然后将报文装入 PPTP 隧道包，最后整个隧道包被嵌入 IP 报文或者其他任何一种通信协议中。在用户看来，这个 VPN 连接和直接通过拨号建立一个 PPP 连接没有什么不同，所以用起来很方便，特别是 Microsoft 已经把它作为标准路由软件的一部分，因此几乎可以在任何系统上使用。不过，因为真正的连接是在 IP 帧上做的，所以在连接认证方面不是非常可靠。

在 Linux 上，纯粹的 PPTP 做法并不多，有一个非标准的方式和 PPTP 很相似，即同样采用 PPP 封包作为通信方式，但是 PPP 封包通过 SSH 的加密通道传送。这个方法比较简单，而且实现起来很容易，详细的内容可以参考王波编著的《freebsd 实用大全》一书。

IPSec 是 IETF（因特网工程任务组）于 1998 年 11 月公布的 IP 安全标准。其目标是为 IPv4 和 IPv6 提供具有较强的互操作能力、高质量和基于密码的安全。IPSec 对于 IPv4 是可选的，对于 IPv6 是强制性的。

IPSec 在 IP 层上对数据包进行高强度的安全处理，提供数据源的验证、无连接数据完整性、数据机密性、抗重播和有限业务流机密性等安全服务。各种应用程序可以享用 IP 层提供的安全服务和密钥管理，而不必设计和实现自己的安全机制，因此减少了密钥协商的开销，也降低了产生安全漏洞的可能性。IPSec 可连续或递归应用，在路由器、防火墙、主机和通信链路上配置，实现端到端安全、虚拟专用网络和安全隧道技术。

需要指出的是，IPsec 还是一个完善中的设计体系，而且往往需要专门的硬件设计。构造一个基于 IPsec 的 VPN 这样的任务，已经超出了本书的范围，详细的内容请参考 IPsec 的有关技术文档。另外，目前有一个基于 IPsec 的免费 Linux 软件 VPN 实现，称为 FreeS/WAN，可以从 www.freeswanorg 下载，可以实现 VPN 的基本支持，可以根据它来实现自己的 VPN 系统。

最后一种与 VPN 有关的协议是 socks V5，是 NEC 开发的，可以和其他加密协议一起实现 VPN 的功能。

5.2.2 网络层

现在来了解一下与 IP 定位/路由、信息传播相关的协议的弱点。路由协议是 Internet 的核心，它用来定位某个数据包应该传送到何处，以及某一台机器位于什么地方。IP 定义了一台机器的身份，而路由路径给出了传送数据报文的路线。遗憾的是，标准 IP 层协议没有对这两件事情做任何的保护。

1. ARP

ARP 即地址解析协议，用来在一个局部网内找到和某个 IP 地址对应的机器。通常我们讨论的都是以太网的 ARP 问题，即由 IP 地址找到对应的 MAC 地址。这是通过广播—应答技术实现的，假设某主机想和与自己同子网的机器通信，例如 202.14.0.3～202.14.0.7，它将使用以太网提供的硬件广播功能，广播一个数据包，其内容包括自己的 IP 地址、MAC 地址及试图通信的 IP 地址。由于硬件广播可以送到以太网电气连接的任何地方，所以 202.14.0.7 如果开着就会接收到这个数据包，于是它向 202.14.0.3 回送一个数据包（因为广播包中已经包含了 202.14.0.3 的 MAC 地址，所以 202.14.0.7 知道如何送数据），内容将包

含 202.14.0.7 的 MAC 地址。这样，一个"广播—应答"就确定了两台机器的 MAC 地址。

为了提高效率，每台机器都会在自己的系统上维护一个对照表，每当得到一个应答或是听到一个地址广播，机器就在自己的对照表中加入一项，这样以后再需要与相应地址通信时就可以直接从表中查到 MAC 地址。因为 MAC 到 IP 的映射可能会改变，所以对照表被设定为每过一段时间更新一次。

当然，这就意味着事实上在同一个子网内任何人都可以通过发送 MAC 地址声明来获得 IP 地址，因此 IP 地址是完全不可靠的。相应地，基于 IP 地址的认证，可能会受到恶意的欺骗。这样的事情在局域网内是难以避免的。

一个减小这种威胁的方法是建立 MAC/IP 绑定机制，这需要在交换机或者路由器上实现。交换机或者路由器维护静态的 ARP 列表同时随时监听 ARP 广播，并对所有可疑信息进行警告，拒绝转发 MAC 地址错误的 IP 报文。

虽然这样的欺骗只能发生在局域网内，但是，由于路由协议的一些问题，上述问题至少在理论上是可以扩展的。

2．ICMP 重定向

TCP/IP 协议系统中，为了能够判断远程主机的可达性和活动状态，定义了一个名叫网间报文控制协议（Internet Control Message Protocol，ICMP），它是 IP 协议的附属协议，IP 层用它来与其他主机或路由器交换错误报文和其他重要控制信息。ICMP 报文是在 IP 数据报内部被传输的，主要用来对系统中的错误进行分析和控制。

ICMP 协议本身并不复杂，常用的 ping 命令就是使用发送一个 ICMP echo 报文检查回应的方式来工作的。为了调试的方便，大部分机器都会回应 ICMP 报文，从而可以用 ping 命令检查它的活动状态。

由于历史的原因，许多系统对 ICMP 包规定了一个尺寸上限，一般是在 64KB 左右。当操作系统接收到 ICMP 报文的时候，需要按照报文标题中的尺寸信息来分配 TCP/IP 缓冲区。如果接收到一个错误或者畸形的 ICMP 报文，其中的尺寸信息是错误的，例如超出了 64KB，缓冲区分配就可能出现问题，严重的情况下会导致 TCP/IP 栈崩溃。这就是 Ping of Death（PoD）攻击。

但是直接发送一个巨大的 ICMP 数据包很容易被路由过滤掉，更简单的办法是直接向主机发送一个 ICMP"碎片"。这个概念大致是这样的：以太网本身必须把数据拆成以太网帧，大小大约是 1.5KB，所以任何大的数据报文都会被拆开，分别封装入各个以太网帧中，这些帧叫做碎片。接到最后一片之后，IP 协议将它们重组成 IP 报文。现在考虑一下，如果发送一个自称是最后一片的 ICMP 碎片会如何呢？很简单，接收主机会试图去重组这个 ICMP 报文，如果碎片来得够多够大，接收主机就会疲于不存在的重组，最终所有 CPU 时间都被耗尽。与前面比较简单的 PoD 不同，这个发送过程几乎不占发送者的带宽，却可以让接收者崩溃。

事实上，ICMP echo 回应除了检查机器是否开着以外，并没有太实际的用处，因此许多系统直接把这个功能关掉，避免意外的问题出现。

Ping of Death 还有一个非常厉害的变形，称为 Smurf 攻击，它基本上就是一个 ping 命令，它的目标地址是一个网络广播地址，而源地址被冒充为将被攻击的目标机器。这样，整

个网络的机器都会向冒充的源地址发送echo回应，而一般来说ICMP echo的优先级高于任何其他服务，那么对应源地址主机就会忙于应付所有的ICMP echo而失去网络响应。

除此之外，ICMP也容易受到其他DoS（拒绝服务）类型的攻击。这是由于ICMP还用来发送路由重定向报文。简单地说，就是用来修正路由表为指向特定主机的数据包提供单独的路由路径的报文。通常客户主机接到这种报文之后，会在自己的路由表中增加一项。当这样的报文非常多的时候，可能导致路由表内存耗尽或者搜索路由表失败。但由于实现的问题，这种技术在Windows系列上较为常见，而在Linux上成功的几率不高。

路由重定向功能还可能导致另外一个攻击手段，即外网机器冒充本地IP，这是通过向路由和目标发送虚假的重定向报文，导致本地机器的路由表被欺骗实现的。不过这种攻击目前很少看到报道。

IP地址欺骗是一个比较复杂的问题，涉及TCP/IP协议的很多问题，同时还涉及Linux系统固有服务的一些设计。简单地说，IP欺骗本身并不是一个安全问题，它仅仅是某一台主机冒充成另外一台主机而已。然而，Linux系统服务中许多服务的身份认证是基于IP地址的。这种情况下，IP欺骗可以成为一种有效的攻击手段。

最重要的情况发生在r命令中，在后面会看到，Linux系统的r命令是基于主机之间的信任关系的，因此IP欺骗的成功就使得进攻者获得了所有的r权限。

3. IGMP组播

IGMP目前还是一个处于实验阶段的协议，提供Internet网际多点传送的功能，即将一个IP包的副本传给多个目标主机。客户机器接收到IGMP消息，标准响应方式是将这个消息报文再传送给别的组内机器，在Windows系列系统中，由于实现上的缺陷，这个过程很容易导致TCP/IP栈崩溃。相对来说，Linux发生这种问题的情况不多。然而即使如此，这仍然是一个容易遭受攻击的弱点，通常用在内核中禁止IGMP组播的办法来消除这个安全性隐患。

5.2.3 传输层

1. UDP拒绝服务攻击

TCP/IP协议定义了两个基本的数据传输协议，一个是面向连接的传输控制协议（TCP），一个是面向无连接的直接数据传送的用户数据报文协议（UDP）。其他任何上层数据传输都只要以其中一个为基础。

单纯传输层的攻击主要是针对这两个协议的堆栈缓冲区，也就是通过发送不合法的或者特殊的数据，以及攻击协议的具体实现，导致系统出错或者崩溃。和前面讲的Ping of Death一样，它们都属于DoS（拒绝服务）范畴。相对来说，UDP协议因为是无连接的，所有的解码和纠错等操作都在应用程序中实现，更加容易遭到拒绝服务攻击。

事实上，拒绝服务攻击是所有攻击方式中最难对付的一种，特别是在传输层的DoS攻击，从原理上说几乎是无法杜绝的。近年来，由于分布式DoS攻击的存在，即使是最简单的DoS攻击也可能打垮一切系统。例如，2000年2月，全球各大ISP几乎同时遭到分布式

DoS 攻击，Yahoo!、新闻网站 CNN、Amazon 和 eBay 等在分布式 DoS 下全部瘫痪，服务器连续十几个小时无法工作，造成高达 12 亿美元的经济损失。

最简单的 UDP 拒绝服务是通过造成 UDP 风暴来完成的，简单地说，就是向目标主机发送无数的 UDP 包，直到对方瘫痪为止。事实上这时候还可能导致另外的效果，即耗尽目标网络中的网络带宽。当然，由于带宽限制，这种攻击一般都使用分布攻击的办法。最典型的例子是 UDP flood，它的做法非常简单：首先取得某台机器的控制权，然后伪造与某一主机的 Chargen 服务之间的一次 UDP 连接，回复地址指向开着 echo 服务的一台主机，当然 echo 服务又会自动回应，这样就生成在两台主机之间的大量的无用数据流，如果生成了足够多的数据流，那么整个网络的带宽都被耗尽。

稍微复杂一点的攻击是使用目标操作系统的缺陷。因为 UDP 是完全使用用户验证的，因此可以用错误或者畸形的 UDP 数据包导致目标机器的验证出错。例如，发送 0 字节 UDP 数据包和 UDP 碎片（参考前面讨论的 ICMP 碎片攻击）都曾经是非常有效的 DoS 攻击方式。其中最出名的是 tear-drop 攻击，简单地说，它也是一种碎片攻击，但是它利用了 Linux/Windows 体系 TCP/IP 栈代码中的一个错误，使得当操作系统重组 IP 包的时候，将包的长度计算为负数，这样就会使得 TCP/IP 栈复制过多的数据进入内核，因为 OS（操作系统）本身不能正确理解负数，而是将负数转换为巨大的无符号数。本质上说，这类攻击只不过是利用了操作系统的漏洞，只要修改相应的代码就可以堵住，但是由于 UDP 处理在 TCP/IP 栈中的实现很复杂，难免有各种各样的漏洞存在，因此 UDP 攻击成为 DoS 中难以对付的问题之一。

2. TCP 拒绝服务攻击

相对来说，TCP 的设计比 UDP 复杂、精巧一些，然而仍然存在一些严重的问题，其中最主要的问题和 TCP 连接建立时的握手协议有关。

正常的一个 TCP 连接需要连接双方进行三个动作，即“三次握手”，其过程如下：请求连接的客户机首先将一个带 SYN 标志位的包发给服务器；服务器收到这个包后产生一个自己的 SYN 标志，并把收到包的 SYN+1 作为 ACK 标志返回给客户机；客户机收到该包后，再发一个 ACK=SYN+1 的包给服务器。经过这三次握手，连接才正式建立。在服务器向客户机发返回包时，它会等待客户机的 ACK 确认包，这时这个连接被加到未完成连接队列中，直到收到 ACK 应答后或超时才从队列中删除。

因为有这样的握手过程，所以攻击者可以构造出非常严厉的攻击手段，其中之一是 synflood。

synflood 又称半开式连接攻击，简单讲就是攻击者制造一个具有虚假源地址的 SYN 握手请求包，于是服务器向对应的源地址发送返回包。但是由于这里的源地址是虚假的，所以服务器不会得到最终的 ACK 确认应答。因此这个连接请求将会被放入请求队列，直到超时。

由于 TCP/IP 请求队列是有限的，如果来的 synflood 请求包太多，整个请求队列就会被这样的等待请求塞满，从而使 TCP/IP 失去响应，这就是所谓的 synflood 攻击。

实际上，在标准 TCP/IP 框架内，synflood 是不可能真正解决的，只有适当更改 TCP/IP 握手方式才能解决问题。最简单的解决思路是在 TCP/IP 队列满的时候随机地抛弃几个尚

未应答的 ACK 请求，这样可以保证系统至少有一个最小响应能力。这种办法被许多操作系统使用。而 Linux 使用了来自 BSD 的另外一个思路，即篡改握手规则。服务器应答 SYN 包时把 SYN 包中的少量数据保存在栈中，在 SYN-ACK 包中把大部分数据传回给发送者，发送者再在 TCP-ACK 中把数据送回建立连接。这个技术可以基本上避免 synflood，但缺点是这样做修改了标准的 TCP 握手准则，可能带来兼容性问题。

使用 syncookies 技术需要在编译内核的时候启用 TCP_SYN_cookies 功能，然后在启动脚本中使用如下的行：

```
echo 1 > /proc/sys/net/ipv4/tcp_syncookies
```

注意：使用 syncookies 可能带来一些意想不到的问题，需要认真阅读内核技术文档。

Land 攻击是另外一种基于 TCP 的 DoS 攻击技术。简单地说，它发送一个源地址与目标地址一样的数据包。在握手请求的 SYN 包中源地址和目标地址都被设置成目标主机地址，这时将导致目标主机向它自己的地址发送 SYN-ACK 消息，结果这个地址又发回 ACK 消息并创建一个空连接，每一个这样的连接都将保留直到超时。这样，很快就可以使 TCP/IP 栈耗尽。这种技术可以使许多 UNIX/Linux 系统崩溃，对于 Windows NT 则会导致系统运行得极度缓慢。解决的方法也很简单，只要增加相应的内核补丁就可以了。

5.2.4 应用层

各种具体的网络应用都可能成为攻击的目标。这里面有两种可能的情况，一种是协议本身就缺乏安全性，另一种则是实现协议的程序可能存在各种各样的漏洞。前者比如 NFS 和电子邮件协议，后者则与具体的编码有关。

1．RPC 和 NFS 相关服务

RPC 即远程过程调用，是 Sun 发明的用来在远程主机上执行特定任务的一种协议。在一般体系中，通过 portmap 和几个 RPC 服务进程实现（如 rpc.statd、rpc.mountd 等）。

由于代码实现的问题，RPC 的几个服务进程很容易遭到远程缓冲区溢出的攻击，相当多的漏洞和 rpc.statd 有关。

使用 RPC 的最重要服务是 NFS，这也是最主要的容易产生安全性隐患的服务之一。

NFS 是 Sun 发明的用来在 UNIX 系统之间共享文件的一种服务协议，实际就是将某个 UNIX/Linux 机器的一个目录共享出来，由其他机器直接使用。共享的动作称为“输出（export）”。例如，host1 机器的 mydoc 目录可以被输出，然后 host2 机器的管理员将它连接到/host1/mydoc 目录下，以后 host2 对这个目录的操作自动转化为对 host1 上相应目录的操作。技术上，通常用 UDP 协议来实现 NFS 的数据传输。

NFS 的主要安全性隐患在于用户权限。一个文件系统被输出的时候，有两种权限需要考虑：标准的 Linux 用户权限和输出权限。输出权限是覆盖在 Linux 用户权限上的，可以把一个文件系统输出成只读或者完全权限，客户机上的用户将按照相应 Linux 用户权限和输出权限的交集来取得权限。

可以这样解释：假设/ftp 目录被输出，我们知道实际上每个目录/文件的属主和权限都

是利用 uid 来定义的，现在假设/ftp 在服务器上的属主是 ftp 用户，其 uid 为 14，当被输出到客户机上时，其 uid 是不变的，所以客户机上 uid=14 的用户将被认为是这个目录的主人。假如目录输出是被设置成完全输出，那么剩下的事情就和在本地机器上存取没什么区别。如果目录输出的时候是只读，那么客户机上所有的用户权限都被限制，即使是 uid=14 的用户也只能获得只读权限。当然，理论上它还是这个目录的主人，只是不可能修改这个目录的只读属性而已。

很显然，由于客户机的 uid 映射是由客户机的/etc/passwd 文件控制的，所以实际上 NFS 的 Linux 用户权限是没有任何实际意义的。你无法禁止客户机使用超级用户权限存取文件，从而任何以 NFS 允许写方式共享出去的文件系统都已经失去了保护，因此，NFS 本身就不是一个安全的共享协议。

考虑到这一点，因此一般情况下，尽量用只读方式共享文件系统，这时对方就总不能修改你的数据。NFS 还有一些和 squash 相关的参数，其中最重要的是 root_squash 和 all_squash。它们可以把对方的根用户（或者所有用户）映射成为本机的 nobody 用户。

NFS 的另外一个问题是，它是基于主机信任的，主机的身份确认完全是由客户机的地址决定。而前面已经指出过，IP 地址本身不是一个可以信任的标志，它可以被伪造，这对于 NFS 是灾难性的。然而，尽管 IP 伪装不算是一个很容易进行的攻击行为，但是要防止它也几乎同样困难，因为它涉及路由器软件。

DNS 的存在使得 NFS 比刚才说的还要脆弱。在 NFS 中可以给出域名而不是 IP 地址（其实即使给出 IP，也还是可能出现问题，因为有域名后缀搜索），而域名到 IP 的解析并不是本地完成的，它是 DNS 服务器的工作。但是 DNS 服务器可能被攻破，从而域名可能指向不正确的 IP。

总之，NFS 中包含着不少隐藏的安全性弱点。正因为如此，任何使用 NFS 的系统都不能说是安全的。

2. r 命令

容易遭受 IP 欺骗攻击的另外一组命令是 Linux 的 r 命令，它们提供了一组不需要给出用户名和口令就可以登录到系统并且执行对应命令的方法，主要有 rsh、rexec、rlogin 和 rcp 等。

它们绕过用户名和口令的方法都是基于远程机器之间的互相信任。在 Linux 中，确定这种互相信任的文件有两种，一种是全局的信任文件/etc/hosts.equiv；另外一种是单个的对于某个用户的信任文件，这个文件存放在每个用户的宿主目录下，名字是.rhosts。

/etc/hosts.equiv 定义了那些远程机器的账号和本地账号等价。例如，这样的行：

```
client.yourdomain.com test
```

代表 client.yourdomain.com 上的 test 账号等价于本地机器上的 test 账号。同样，client.yourdomain.com cook 代表 client 上的 cook 账号等价于本地账号 cook。

单个信任文件.rhosts 存放在用户的宿主目录下，格式与 hosts.equiv 类似，所不同的是，这个文件可以用来定义不同的用户账号之间的等价性。因为其位置是在用户的宿主目录下，

所以不需要给出等价的本地账号，而文件中写出的账号名则代表远端机器的账号。例如，在server的user1用户的宿主目录（/home/user1）下建立如下的.rhosts文件：

```
host1.yourdomain.com user1
host2.yourdomain.com user2
```

这样的文件意味着除了host1上的user1用户被等价为server的user1用户之外，host2上的user2用户也被等价为server的user1用户。

不难看出，前面讨论的NFS安全性隐患，r命令也全都有。除此之外，r命令还有另外一个严重的问题：它可以用来传递命令，并且传送过程中不加密（NFS也基本是明文传送，不过相对来说，截听NFS不是很有价值）。

所有的这些原因导致了r命令成为Linux系统中最不可信赖和最不安全的命令组。庆幸的是，和无法完全避免的NFS不同，r命令不是不可替代的，SSH几乎可以完全完成r命令的功能，同时很好地解决安全性问题。

3. 电子邮件：炸弹和身份认证

电子邮件是Internet中最让人困惑的部分——虽然WWW很复杂，但是远不及电子邮件。除此之外，事实上sendmail的问题和麻烦很多，虽然有了一些很好的sendmail代替程序，但是SMTP协议本身的问题一直没有真正解决。

单从理论上说，SMTP协议并不复杂，它就是一次次邮件接力的过程。当一台计算机准备发送电子邮件的时候，它连接到服务器的25端口，然后开始一个邮件会话，这个过程中客户机必须给出发送者和接收者的邮件地址。如果服务器认为这两个地址是“可以接受的”，那么客户机将邮件内容编码成ASCII码传送给服务器，服务器将它保存在自身的投递队列里面，这样一次邮件接力就完成了。

称为邮件接力的原因是，服务器接下来要对邮件进行具体处理。邮件的目标地址通常是服务器的一个账号，那么邮件服务程序将会调用预设的邮件分拣程序，将邮件送入这个账号的邮箱中，这样投递才完成。也可能邮件的目标地址不在本地，那么邮件服务程序需要解析邮件地址得出对应邮件地址的服务主机，然后再一次邮件接力将它投递到对应的服务主机——这次的投递由服务器完成。也许，对应主机还要进行下一次投递接力，这样连续地投递直到送到最终用户的邮箱中。

由于邮件是这样经过重重接力投递过去的（当然，正常情况下最多两次），所以大部分主机都要完成不属于自己的邮件的投递工作。这可以分成几种情况，最简单的情况是主机根本不考虑邮件的发送和接收地址，只要送到它的25端口的信件就给予投递。这种计算机叫做open relay，这种主机通常都会被大量的垃圾邮件淹没。除此之外，绝大部分邮件主机都必须对邮件的发送/接收地址、客户机的IP等做一些限制，避免垃圾邮件的攻击。

SMTP协议对于发送邮件本身是不做任何验证的，换句话说，任何人都可以对你发信，而且邮件服务程序总要把邮件放入你的邮箱，所以，如果谁愿意让你的邮件报废的话，简单地给你发送一百万封空邮件就可以了。这个方法其实连攻击都算不上，不过非常行之有

效，标准的术语叫做“邮件炸弹”。

除此之外，电子邮件还有一些技术含量不高却很令人头痛的问题，例如垃圾邮件。不过这些跟安全性并没有什么关系。相反，有一个问题很容易让用户头痛，就是身份认证的问题。

标准 SMTP 协议对于发送邮件并不需要登录到系统，这使得任何人都可以冒充系统本地用户来发送电子邮件，当然就本质来说这个问题不能算是问题，邮局也不可能禁止发信者送出匿名信。不过在今天，电子邮件已成为极重要的交流手段，很可能出现意料之外的要求。特别是为了在一定程度上限制垃圾邮件，发明了 ESMTP 协议，它可以设置为发送邮件的时候进行身份认证，给出用户在系统中的账号和密码。当然，这只能是在一定程度上有用，真正的解决方法，还要使用 PGP 和数字签名。

从安全性问题考虑，邮件系统除了 SMTP 固有的不安全因素之外，主要的和本地投递有关，而最容易出现问题的是 sendmail。

当邮件地址是本地账户的时候，邮件服务程序就需要进行本地投递。投递方式依赖于邮件服务程序的具体实现。不过，几乎所有的邮件服务程序都会支持投递到程序，即通过管道将邮件内容传送给其他程序。如果服务程序设计得不够周全，很容易导致系统安全问题。

sendmail 作为最主要的邮件服务程序，安全问题也是最多的。它的主要问题是所有的功能被集中在一个大程序中，为了完成所有的功能，这个程序必须 suid 到 root 执行。这样，一旦被溢出，攻击者就会取得 root 权限。另外，sendmail 本身不处理最终的邮件投递，大部分本地投递工作由 procmail 或者 UNIX mail 程序完成。需要投递到程序的邮件则通过一个 shell 来启动对应程序，这也额外地增加了危险性。

sendmail 可以通过将本地 shell 改成一个名叫 smrsh 的邮件专用 shell 来完成，它基本上和标准 shell 的功能相同，但是加入了为邮件系统做的安全性考虑的功能，例如禁止执行 suid 程序等。遗憾的是，并不是所有程序都可以这样处理，例如著名的邮件列表程序 majordomo 由于必须 suid，就无法使用 smrsh 来运行。

比较现实的方法应该是尽量更换不怎么可靠的 sendmail 程序，使用相对更安全的邮件服务程序。目前主要的替换产品有 postfix、qmail 和 exim。其中，postfix 和 qmail 都是设计得比较全面的邮件服务系统，具有比 sendmail 更好的性能和安全性。特别是它们在设计时都考虑了尽量避免不必要的 suid 到 root，使得被溢出的危险大大降低。在某些情况下，还可以使用 chroot 方式将所有邮件程序锁在特定的目录中，这样即使被溢出，危险也要比 sendmail 小得多。

与邮件相关的还有两个主要的协议：POP（邮局协议）和 IMAP（Internet 邮件存取协议），用来让客户机从服务器邮箱内取得邮件内容。

由于邮箱实际是属于用户的一个文件，访问它需要进行用户身份认证，因而一般情况下，这个认证是个简单的校验用户名/口令的过程。不过标准的 POP/IMAP 口令认证都是明文传输的，很容易被监听到，因此发明了用于 POP 的加密传输协议，这个协议的一个主要实现是 qpopper 程序，它可以提供一些高级的选项，特别是支持以加密方式在客户和服务

器之间转发口令字，以及对 pop3 使用与账号登录不同的认证口令等。

4. DNS

Linux 的 DNS 服务程序——bind 也是著名的容易受到攻击的程序。而且，因为 DNS 本身极端重要，一旦 DNS 出现问题，后果将变得非常严重。

DNS 的功能是把 Internet 域名转换为 IP 地址，这在网络中是至关重要的。

（1）几乎所有的主机信任关系都是基于域名而不是 IP 地址，以便可以透明地修改。

（2）如负载均衡之类的工作，通常利用 DNS 数据系统的多记录随机分配来完成，这种情况下，无法直接使用 IP 地址来访问主机。

（3）DNS 是分级体系，一旦某个 DNS 出现问题，整个下辖系统全部会出现问题，而且因为 DNS 缓存的问题，错误会在 Internet 上扩散。

（4）DNS 服务程序因为运转在 TCP/UDP 的 53 端口——小于 1024 的特权端口，因而必须用超级用户身份执行，而它要读配置文件和写大量的 log 记录，也要求具有存取系统文件的能力，因此一旦被溢出，后果是灾难性的。

对于 DNS 的攻击，首先是 DoS。一般情况下，DNS 服务器是在防火墙系统的外边，对查询数据流不做防护以方便客户机的访问。但这也给攻击者带来了方便。

（1）一个没有防火墙的系统，很容易发生流量过载，这种问题甚至和任何具体实现无关，就是简单的路由超载。

（2）DNS 服务程序 bind 因为是沉默地在 53 端口监听并且对一切访问做出回应，需要耗费大量的内存和 CPU 时间。

通常一个部门的主 DNS 服务器必须支持整个部门/公司的 DNS 访问，这种中心 DNS 负载量是相当大的，经常会消耗大量的内存空间。而 DNS 主要的工作是在 UDP 53 端口，所以用精心设计的 DoS 程序可以很容易地将其打垮。特别是，DNS bind 程序是作为长期使用的程序，在某些情况下设计不周全的代码会带来资源泄漏，这种情况下，系统拒绝服务只是个时间问题而已。

DNS 被篡改带来了另外一种可能的“IP 欺骗”，它和 IP 欺骗的技术是不同的，但是效果基本上没什么区别。因为主机信任关系经常会被写成用域名代表主机的形式，这样如果域名记录被篡改了，那么主机信任关系就此被攻破。

除此之外，域名篡改还有另外一个不是很高明但是非常有效的用处，就是让原来访问某台主机的数据流流向另外一台主机。这种办法最有效的使用是篡改 WWW 站点的域名记录，让原来指向这个站点的访问重定向到攻击者的站点，从而骗取敏感信息。这类域名篡改技术有个听起来比较专业的名字——DNS 毒药。

DNS 毒药实现起来不是很容易，但是有一些技术可以帮助攻击者达到目的。其中最惯用的办法是 DNS ID hacking。简单地说，当 DNS 服务器接收到一个不属于自身管辖范围的域名记录的 DNS 解析请求时，服务器将和上层或者根服务器联系，取得这个域名的 IP，并且缓存在自己的缓存记录中。攻击者可以监听 DNS 报文，抢在目标 DNS 服务器回答之前“抢答”DNS 请求报文。DNS 服务程序并不检查应答报文是谁发出的，只要应答报文头

部的报文 ID 和请求报文一致，就认为是对自己的应答，因此攻击者需要的全部任务只是猜测一个正确的报文 ID 就完成了。因为 DNS 报文是明文传送的，所以更有效的方法是简单监听报文并且从请求报文中取得 ID，这样 DNS 攻击就成功了。以后，因为这个应答记录被缓存，马上整个系统都会中毒。

就算技术上的问题让你不能完全监听 DNS 报文，那也没关系，因为 DNS bind 的实现比较弱：第一次查询使用了一个随机 ID，下一次查询一定是这个 ID 加 1，依此类推。Windows NT 的实现则更弱，它每次查询的 ID 都等于 1。

但是没有什么非常有效的手段对付 ID hacking，因为 DNS 报文不能加密。

DNS 的下一个传统问题是 zone transfer（区域传递）。区域传递就是把本 DNS 服务器负责的一个完整域内的所有 DNS 记录直接传送给客户机。但是这样做在安全性上却很危险，因为它会泄露太多的关于本地网络的信息，如果攻击者拿到了所有 DNS 记录，就可能反向判断出网络的整体拓扑结构，并且找到进行攻击的薄弱环节。

除此之外，由于一般情况下 DNS 服务器处于防火墙外面，相当于一台裸露的 Linux 主机，而且它通常会具有穿越防火墙的能力，一旦攻击者攻破了 DNS 机，就会带来一个进入防火墙内部的跳板。

传统上，bind 程序是一个著名的容易被攻击的程序，许多致命的系统漏洞和它有关。原因是因为它用文本串的方式传递 DNS 报文，并且使用标准的 C 库函数和格式化字符串功能，很容易被溢出。而且，由于它运行在超级用户权限，一旦被溢出，后果就是灾难性的。

解决 bind 的缓冲区溢出问题，除了不断升级系统之外，就只有用 chroot 或者 LIDS 的方案，让 bind 程序不能执行 shell，从而即使溢出，攻击者也拿不到 root shell。chroot 的方法比较简单，也是比较传统的解决方案，named 进程本身就有一个参数用来让自身进入 chroot 模式。

命令格式

```
named -t [目录]
```

不过，由于 named 需要访问许多文件，所以这些文件必须被一起复制到对应的 named 工作目录下面，特别是 syslog。详细内容可以参考 bind-chroot-HOWTO。

5. FTP 和 WWW

作为最主要的 Internet 服务（除了 SMTP 以外），FTP 和 WWW 受到攻击的次数是非常多的，然而总的来说，在这个问题上，可以讨论的内容并不多，除了前面提到的用来对付监听的 SSH 和 SSL 之外，就都是些老生常谈的内容。这是因为这两个服务经过太多的攻击和堵漏的结果，目前的攻击手段几乎都是和具体的服务器程序实现有关。

1）FTP

对于 FTP 服务器，需要考虑的是，未经授权的用户禁止在服务器上进行 FTP 操作。匿名 FTP 用户不能读取未经系统所有者允许的文件或目录。未经允许，匿名 FTP 用户不能在

服务器上建立/删除文件或目录。这些功能的具体实现几乎都和实际的服务器软件实现有关。

目前有几个主要的FTP服务程序，最常见的是wu-ftpd、proftpd和ncftpd。由于wu-ftpd的性能比较好，使用的系统最多，但是传统上它的安全性问题也最多。

wu-ftpd用系统的标准用户来验证用户身份，匿名用户被映射到系统账号中的FTP用户。另外，凡在/etc/ftpusers文件中出现的用户都将被服务器拒绝提供FTP服务（这个功能是直接编码到wu-ftpd中，但是也可以通过PAM实现）。服务器管理可以建立“不受欢迎”的用户目录，拒绝这些用户访问。FTP目录安全设置如表5-10所示。

表5-10 FTP目录安全设置表

目　录	描　述
FTP主目录	这个目录的所有者应该设为FTP，并且将属性设为所有的用户都不可写，防止不怀好意的用户删改文件
FTP/bin目录	该目录主要放置一些系统文件，应将这个目录的所有者设为root（即超级用户），并且将属性设为所有的用户都不可写。为保证合法用户可显示文件，应将目录中的ls文件属性设为可执行。近来的wu-ftpd服务程序允许将ls命令内嵌到ftpd之内，这样可以删除ls文件提高可信度
FTP/etc目录	将这个目录的所有者设为root，并且将属性设为所有的用户都不可写；将目录下的group文件和passwd文件的属性设为所有用户只读属性，并用编辑器将passwd文件中用户加过密的口令删除
FTP/pub目录	将这个目录的所有者置为FTP，并且将它的属性设为所有用户均可读、写、执行

只有在服务器的/etc/passwd文件中存在名为FTP的用户时，服务器才可以接受匿名FTP连接，匿名FTP用户可以使用anonymous或FTP作为用户名，自己的Internet电子邮件地址作为保密字。当匿名FTP用户登录到FTP时，wu-ftpd通过执行chroot将匿名用户锁在/etc/passwd中FTP用户的宿主目录中，因此，匿名用户的安全问题主要集中在这个目录的存取权限上。

除此之外，许多系统还会给出一个incoming目录用于让匿名用户上传，这个目录的属性需要设置为所有人都可以存取和读写，这是比较容易出现问题的地方。可以利用wu-ftpd的功能将它设置成一旦上传就不能删改，增加系统的可靠性。另外，还需要利用记录程序/var/log/xferlog记录各种登录和连接信息。

除了wu-ftpd之外，proftpd和ncftpd的基本设置思路也可以参考上述内容，总之都要控制匿名用户的访问。另外，FTP程序的缓冲区溢出和core dumped问题都可能带来安全性漏洞，请参考有关的资料。

2）HTTP

HTTP协议是最主要的网络信息传输协议。关于它的主要问题，前面在讲SSL的时候已经讨论过了，它本身是明文传送，容易遭到监听的威胁。

除了这种纯粹的监听和IP欺骗问题之外，HTTP服务器还会有一些特有的安全性问题。其中最重要的和CGI程序有关。

CGI即通用网关接口，是一种用来帮助服务器和用户交互的程序。它运行在服务器上，

接收客户程序的信息，进行必要的操作，然后将运行结果返回给用户。最典型的例子是留言板，它从客户 html 输入中取得输入信息，写入到服务器上的文件中。广义地说，如 ASP、PHP 等服务器端脚本也属于这个范畴。

由于 CGI 程序具有访问服务器端资源的能力，因此小心控制 CGI 程序的权限是非常重要的。这里面有一些隐含的问题。

目前 Linux 下最主要的 WWW 服务程序是 apache。因为页面服务程序总是需要服务器文件的能力，所以需要设置 apache 的运行权限。为了安全起见，apache 使用 bind-fork 方式。apache 首先使用超级用户权限启动，获得对 80 端口的控制后，fork 自身将子进程切换到实际的用户（一般是 nobody），然后用子进程应答 HTTP 请求。这个动作可以避免被缓冲区溢出到 root shell，即使溢出了，攻击者也只能得到 nobody 权限。

但是对于 CGI 程序，因为 CGI 程序通常要写服务器端文件，就必须考虑写权限的问题。一种办法是把需要写的文件设置成 nobody 可写或者是任何人都可以写，但这个办法显然不值得推广。另外一个办法是使用 suid。将 CGI 程序 suid 到对应的用户，这样无论 apache 程序的权限是什么，CGI 程序都自动获得用户权限。

除了这类方式外，apache 还存在其他一些控制权限的方式，最重要的是 suEXEC。这是一种按照在 httpd.conf 或者对应目录的权限设置 CGI 用户权限的机制。suEXEC 允许下面两个功能：

- 对于虚拟主机，允许每个 VirtualHost 段落使用自己的 user 和 group 子句。
- 对于 http://www.yourdomain.com/～someone 这样的主页服务，允许每个用户在自己的目录下设置一个 CGI 目录，处于这个目录中的 CGI 程序自动获得对应用户的权限。这两者对于虚拟主机和多用户的主页机器是非常重要的。

在一般情况下，尽量使用 suEXEC 功能可以获得比较好的安全防护。另外，在迫不得已要 suid 到 root 的情况，尽量用 PHP 或者 Perl 这样的脚本语言。它们通常会有额外的完全防护，以便尽量减少被缓冲区溢出的可能性。Perl 有个称为 suidPerl 的 apache 专用版本，用来提高脚本的安全性。

另外一种对 apache 威胁很大的攻击方式是 DoS。因为 apache 本身的复杂性和设计问题，远程攻击很容易将安装 apache 计算机的内存耗尽。即使没有 DoS，WWW 服务的负担也很容易让 apache 陷入困难之中，这已经不是简单的安全性问题了，而需要综合考虑很多方面才能解决。

6．inetd

UNIX/Linux 系统有一类独特的网络服务进程，称为 inetd。这里用 inetd 程序来统称这种提供 inetd daemon 功能的程序，包括 inetd、xinetd 和 tcpserver 等。

一般情况下，要建立一个 TCP 服务程序，至少需要做下面的几项工作。

首先，要建立一个监听用的 socket，把它绑定到服务器的某个端口上，然后让它去监听用户的连接请求。

一旦有客户连接到达，服务进程 fork 复制出自身的一个副本，然后把连接的 socket 交

给子进程，自己继续监听。子进程现在可以用 read 和 write 对 socket 进行操作，完成服务功能。

上述过程是比较复杂的，而 inetd 服务进程的功能就是把上面的过程加以简化。它监听服务端口，一旦有请求到来，inetd 启动对应的服务，但是并不把 socket 句柄传递给对应的服务程序，而是接管服务进程的输入输出，由客户机送来的请求数据，格式化后传递给服务程序的 stdin。而如果服务程序想要向客户机发送数据，只要直接向 stdout 写就行了。这样，socket 编程被简化成了 scanf/printf 的操作。

因此，许多服务被设计成通过 inetd 方式来完成服务，当然这就意味着系统中必须有一个 inetd 服务程序，最早的这种服务程序就是 inetd。

目前来说，inetd 的漏洞并不是很多，但这仅仅是因为大部分著名的漏洞都被堵上了。要知道，inetd 支持许多 TCP 服务，而且必须工作在 root 权限下，所以一旦有漏洞就是致命的。

inetd 的原始设计仅仅是个转换工具，几乎没有任何附加的安全功能，特别是没有对用户的 IP 地址进行过滤和记录用户连接的功能。为此，有人开发了一个称为 tcpwrapper 的附加工具包，它可以通过/etc/host.allow 和/etc/host.deny 对客户机的 IP 地址进行限制。不过，目前更流行的做法是用其他的服务程序替换 inetd，最主要的是 xinetd 和 tcpserver。这两者都可以替代 inetd 的功能，但是具有更好的安全性和网络性能。

7．端口扫描

扫描本身不算一种攻击行为，但是它常常可以成为攻击发起前的准备工作。扫描器能够自动检测远程或本地主机的安全性弱点，发现远程服务器各种 TCP 端口的分配、提供的服务及相应的软件版本，记录目标给予的回答，搜集关于目标主机的各种有用信息。扫描器可以帮助发现目标主机存在的一些问题，而这些问题可能恰恰就是黑客攻击的关键点。

端口扫描也是一种获取主机信息的有效方法。在 UNIX/Linux 系统中，任何用户均可使用端口扫描程序而不需要 root 权限。一般情况下，运行 UNIX/Linux 操作系统的主机在小于 1024 的端口提供了很多服务，包括许多特有服务。使用 telnet 命令连到这些端口都会得到系统的响应。如果利用端口扫描程序扫描网络上的一台主机，就可以清楚地知道这台主机运行的是什么操作系统，提供了哪些服务。因为一般的 Windows NT 除了在 21、80 端口监听以外，还在 135、139 等端口进行监听，而 Windows 98 通常只在 139 端口进行监听。因此，从扫描的端口数目和端口号能够判断出目标主机运行的操作系统，通过收集扫描的信息，也能够轻松地掌握局域网络的构造。

表 5-11 所示为一些常用端口号和对应服务的对照表，不过应该认识到，这种对应仅仅是约定，特别是对于高于 1024 的端口，并没有严格的规范进行约束。

表 5-11　常用服务端口对照表

服　务	端　口	服　务	端　口
socks	1080/tcp	wins	1512/tcp
socks	1080/udp	nfs	2049/tcp　nfsd 2049/udp　nfsd

续表

服 务	端 口	服 务	端 口
mysql	3306/tcp 3306/udp	gopher	70/tcp 70/udp
netstat	15/tcp	finger	79/tcp 79/udp
linuxconf	98/tcp	http	80/tcp 80/udp
rndc	953/tcp 953/udp	pop3	110/tcp 110/udp
squid	3128/tcp	imap	143/tcp 143/udp
ftp	21/tcp 21/udp	ldap	389/tcp 389/udp
ssh	22/tcp 22/udp	rtsp	554/udp
telnet	23/tcp 23/udp	shell	514/tcp
smtp	25/tcp 25/udp	syslog	514/udp
nameserver	42/tcp 42/udp	uucp	540/tcp

关于其他端口扫描的相关内容，请参见 5.3 节“安全工具”。

5.3 安 全 工 具

就如这个世界有矛就有盾一样，网络世界有各种各样的攻击工具，也有各种各样的安全工具。这里只给读者介绍一些 Linux 下较常见的安全工具，与 Linux 本身类似，这些工具大多也是开放源码的自由软件，恰当地使用它们，可以提高系统的安全性。

5.3.1 tcpserver

tcpserver 是一个 inetd 类型的服务程序，它监听进入的连接请求，为要启动的服务设置各种环境变量，然后启动指定的服务。tcpserver 允许限制同时连接一个服务的数量。当服务过忙时，inetd 具有一种连接速率限制机制，以临时停止服务。

1. tcpserver 程序的语法

```
tcpserver [opts] [host][port] [prog]
```

- opts：一系列选项。
- host：服务器名称。

- port：服务器端口。
- prog：tcpserver 等待来自 TCP 客户端的连接，每个连接都运行 prog 程序。每个连接同时也建立若干环境变量（有关环境变量的资料可参阅 http://cr.yp.to/ucs-pi-tcp/environment.html），服务器的地址由 host 加 port 唯一表示。port 可以来自/etc/services 或数字，若 port 为 0，tcpserver 将选择一个可用的 TCP 端口。主机可以是 0，这时 tcpserver 允许来自本地任何 IP 的连接。host 也可以是一个 IP 地址，这时 tcpserver 仅接受来自该 IP 地址的连接。host 还可以是主机名，主机名使用 dns_ip4 认证反馈。当接收到 SIGTERM 时，tcpserver 退出服务。

2．可选项

可选项包括综合选项、连接选项和数据集选项，分别如表 5-12、表 5-13 和表 5-14 所示。

表 5-12　综合选项表

选　项	描　述
-q	不打印出错消息
-Q	（默认）打印出错消息
-v	打印出错消息和状态信息

表 5-13　连接选项表

选　项	描　述
-c n	当同时连接的数目超过 n 时，不再处理其他连接。若有 n 个 prog 程序副本同时运行，新的连接请求将被延迟到一个副本结束运行时才接受。n 必须是正整数，默认值为 40
-x cdb	遵循使用 tcprules 编译进 cdb 的规则，这些规则可以指定环境变量的设置和拒绝来自错误的数据源的连接；tcpserver 正在运行的时候，可以返回到 tcprules 修改这些规则
-X	即使 cdb 不存在也允许连接，这是与-x cdb 不同的地方。通常情况下，tcpserver 会断开 cdb 不存在的连接
-B banner	在每个连接建立之后，tcpserver 会在搜索＄TCPREMOTEHOST、＄TCPREMOTEINFO 和检验 cdb 之前，立即将 banner 写入到网络
-g gid	在准备好接受连接后将组 ID 切换到 gid，gid 必须为正整数
-u uid	在准备好接受连接之后将用户 ID 切换到 uid，uid 必须为正整数
-U	和-g＄GID -u＄UID 一样，但＄GID 和＄UID 由 envuidgid 设置
-1	在准备好接受连接之后，将本地端口号在标准输出设备上打印出来
-b n	允许预定约 n 个 TCP SYNs，在某些系统，n 被预设为 5。在支持 SYN cookies 的系统，与 backlog 无关
-o	不修改 IP 选项，若客户端通过某个 IP 源路由发送信息包，信息包将被按原路由送回
-O	（默认）不采用 IP 选项，客户端仍然可用源路由连接和发送数据，但信息包通过默认的路由线路送回
-d	当远程主机响应缓慢时，将数据的发送延迟若干秒
-D	不延迟数据的发送，使 TCP_NODELAY 有效

表 5-14　数据集选项表

选　　项	描　　述
-h	（默认）在域名服务器中搜索远程主机名，设置环境变量$TCPREMOTEHOST
-H	不在域名服务器中搜索远程主机名，删除环境变量$TCPREMOTEHOST。为避免产生回路，必须将该选项应用在服务器的 TCP 端口 53
-p	在域名服务器搜索远程主机名之后，搜索该主机的 IP 地址。若没有与客户端 IP 地址一致的地址，则删除环境变量$TCPREMOTEHOST
-P	（默认）不做替代，即如果不匹配，也不改变那个数量
-l localname	不在域名服务器搜索本地主机名，将环境变量$TCPLOCALHOST 设置 localname，通常 localname 被置为 0。为避免产生回路，必须将该选项应用在服务器的 TCP 端口 53
-r	（默认）从远程主机获取环境变量$TCPREMOTEONFO
-R	不从远程主机获取环境变量$TCPREMOTEONFO，为避免产生回路，必须将该选项用在服务器的 TCP 端口 53 或 113
-t n	在尝试$TCPREMOTEINFO 的连接 n s 后放弃尝试，默认值为 26

5.3.2　xinetd

xinetd（Extended Internet Services Daemon）与 inetd 非常相似，较之于 inetd 又更强大更安全。目前许多发行版本带有 xinetd 程序。如果提供的服务比较简单而且负担不重，那么 xinetd 是一个合适的选择。

xinetd 具有下述特点：

- 支持 tcp、ucp 和 RPC 服务。
- 基于时间段的访问控制。
- 功能完备的 log 功能，可以限制 log 文件的大小。
- 可有效地防止 DoS 攻击。
- 可以限制同时运行的同类服务器数目。
- 可以限制启动的所有服务器数目。
- 在特定的系统端口绑定某个服务，从而实现只允许私有网络访问该服务。
- 可以作为其他系统的代理。

1．xinetd 的安装

（1）直接使用系统自带的 xinetd。

（2）自行编译安装。

- 首先从 www.xinetd.org 下载得到 xinetd。
- 然后在 linux 环境下编译安装#./configure；make；make install。

configure 可以带有的参数如表 5-15 所示。

表 5-15　configure 参数表

参　　数	描　　述
–with-libwrap	使 xinetd 通过查看 tcpd 配置文件（/etc/hosts.{allow，deny}）进行访问控制，要利用该功能，系统必须安装 tcp_wrapper 和相关库
–with-loadavg	使 xinetd 处理 max-load 配置选项，从而在系统负载过重时关闭某些服务进程以防止某些 DoS 攻击
–with-inet6	使 xinetd 支持 IPv6

2. xinetd 的配置说明

xinetd 的默认配置文件是/etc/xinetd.conf，可以通过在 xinetd.conf 里面添加 includedir/etc/xinetd.d，而把相应的服务配置文件分开存放到/etc/xinetd.d/下。

```
#/etc/xinetd. conf文件
#
#Simple configuration file for xinetd
#
#Some defaults, and include/etc/xinetd. d/

defaults
 {
      instances          =60
      log_type           =SYSLOG authpriv
      log_on_success     =HOST PID
      log_on_failure     =HOST
      cps                =2530
}

includedir /etc/xinetd. d
#
-----------------------------------------------------------------------
#/etc/xinetd/time文件
#
#default: off
#description: An RFC 868 time server. This is the tcp\
#version, which is used by rdate.
service time
 {
      disable=yes
      type           =INTERNAL
      id             =time-stream
      socket_type    =stream
      protocol       =tcp
      user           =root
      wait           =no
}
#
#/etc/xinetd/time文件到此结束
```

通过上面的例子可以看到，xinetd 文件的格式为：

```
service <service _name>
 {
       <attribute> <assign_op> <value> <value>…
              …
}
```

- service：必需的关键字，服务由 service_name 定义。
- service_name：通常是标准网络服务名，也可以增加其他通过网络请求激活的非标准服务。
- attribute：文件属性。

attribute 的属性如表 5-16 所示。

assign_op 可以为=、+=或-=。=的作用是分配一个或多个值，所有属性均可使用；+=或-=的作用分别是将其值增加到某个现存值表或从现存值表删除，某些属性可以使用。

- value：给定属性设置参数。

表 5-16　attribute 属性表

属　性	描　述	允 许 值
access_times	设置服务的可用时间间隔	格式：hour：min-our：min
bind	把一项服务绑定到机器的特定端口	bind=（接口的 IP 地址）
cps	限制进入的连接率	
disable	禁用服务	yes（禁用）/no（启用）
enabled	允许的服务名列表	
env	以 name=value 形式列出的串	
flags		DISABLE：指定禁止的服务，级别高于 ENABLED，即使 ENABLED 某个服务设置了 DISABLE 属性，服务仍然会被禁止。 IDONLY：只在远端识别远程用户时才接受该连接，该标记只适用于基于连接的服务，若未使用 USERID log 选项则该标记无效。 INTERCEPT：截获包或允许的连接以确认其是否来自于允许的位置(INTERNAL 服务和多线程服务不能被截获)。 NAMEINARGS：通过设置 server 中的 tcpd 和 server_args 中的服务器程序名，可以使用 tcpd。 NODELAY：若服务为 tcp 服务，且设置了 NODELAY 标记，则 TCP_NODELAY 标记将被设置在 socket 上，若不是 tcp 服务则该操作无效。 NORETRY：若 fork 失败不重试
group	设置服务进程的 gid	组名必须存在于/etc/group；若不指定，将使用用户的组（/etc/passwd）；若 xinetd 的有效 uid 不是超级用户，该属性无效默认情况下和服务名相同
id	用来唯一地指定一项服务（有些服务的区别仅仅在于使用不同的协议，因此需要使用该属性加以区分）	
include	以/etc/xinetd/service 形式取得文件名，该文件将认为是一个新的配置文件	
includedir	以 includedir/etc/xinetd.d 形式取得文件名，每个文件都包含在该目录中，被认为是 xinetd 的配置文件	
instances	设置可同时运行的最大服务器程序数	任意值或 UNLIMITED，UNLIMITED 意味着无限
log_on_success	指定服务器程序启动应记录的信息和服务器程序结束的时间	DURATION：记录服务器程序会话持续时间。

续表

属　性	描　述	允 许 值
log_on_success		EXIT：记录服务器程序终止时进程终止的状态和信号等事实。 HOST：记录远程主机地址。 PID：服务进程的PID。 USERID：通过RFC1413协议记录远程用户的UID，只可用于多线程流服务
log_on_failure	指定服务器程序启动失败时记录的信息	ATTEMPT：记录一次失败的尝试。 HOST：记录远程主机地址。 USERID：通过RFC1413协议记录远程用户的UID，只可用于多线程流服务
log_type	指定服务日志输出位置	SYSLOG：日志输出到指定设备的系统日志。 file：日志被追加到文件，若不存在则建立文件
max_load	服务达到最大允许连接的负载值	
nice	指定服务器程序优先权	
no_access	指定被拒绝特定服务的远程主机	
only_from	指定可获得特定服务的远程主机，若不指定值，则任何人无法获得该项服务	取值如表5-17所示
passenv	xinetd 环境中传递给服务器程序的环境变量列表	
port	定义服务端口号	若该服务在/etc/services 中列出，则必须与services中列出的端口号相等
protocol	指定服务使用的协议	该协议必须在/etc/protocols中定义，若不指定，将使用服务的默认协议
redirect	把tcp服务重定向到另一个主机，该属性优先权高于server属性	redirect=（IP地址）（端口）也可以用主机名代替IP地址
rpc_number	指定未列表的RPC服务数	
rpc_version	指定RPC服务的RPC版本号	
server	指定执行该服务的程序	
server_args	指定传送给服务器程序的参数，不能包含服务器程序名	
socket_type		stream（基于流的服务）、dgram（基于数据报的服务）、raw（需要直接IP访问的服务）和seqpacket（需要可靠的连续数据报传输的服务）
type		RPC（RPC服务）、INTERNAL（由xinetd自身提供的服务）和UNLISTED（没有列在标准系统文件中的服务，如/etc/rpc或/etc/services）
user	设置服务进程的uid	用户名必须存在于/etc/passwd，若xinetd的有效uid不是超级用户，该属性无效
wait	设置服务的类型是单线程还是多线程	yes：单线程服务 no：多线程服务

表中给出的属性并非都要在系统中指定。通常一个服务必需的属性有下面几项：

socket_type
user　　　　（只用于非内部服务）
server　　　（只用于非内部服务）

wait
protocol　　（只用于 RPC 和未列出的服务）
rpc_version　（只用于 RPC PC 服务）
rpc_number　（只用于未列出的 RPC 服务）
port　　　　（只用于未列出的非 RPC 服务）
下面的属性支持所有的操作符，其取值如表 5-17 所示。
only_from
no_access
log_on_success
log_on_failure
passenv
env　（不支持"-="）

表 5-17　only_from 的取值

格　式	描　述
%d.%d.%d.%d	全 0 代表网络，例如 128.138.12.0 匹配 128.138.12 子网的所有主机，0.0.0.0 匹配所有的 Internet 地址，IPv6 主机可以 abcd: ef 01::2345:6789 的形式指定，IPv4 地址没有必要这样做
%d.%d.%d. {%d，%d，…}	地址分解形式，并非一定需要 4 部分，例如%d.%d. {%d，%d，…，%d} 也可以，但是被分解的部分必须是整个地址表示的最后一部分，这种形式不适用于 IPv6 主机
网络名（来自/etc/networks）	这种形式不适用于 IPv6 主机
主机名	与 xinetd 建立连接时，返回的权威名（DNS 反向解析出来的名字）将与指定的主机名相比较，也可以使用.domain.com 形式的域名，若返回的客户机 IP 在.domain.com 内，则匹配
IP 地址/网络掩码	格式为 1.2.3.4/32；IPv6 的地址/网络掩码合法格式为 1234::/46
不指定	对任何地址都禁止

xinetd 的内部服务（包括基于流和基于数据报的服务）有 echo、time、daytime、chargen 和 discard。除了不需要 xinetd 为之建立另外的进程的服务外，上述服务与所有的其他服务一样有相同的访问限制。前者（time、daytime、基于数据报的 echo、chargen 和 discard）则没有 instances 数量的限制。

xinetd 同时提供两种基于流的未列表的内部服务：服务器程序（server）和服务（services）。前者列出服务器程序运行的信息，后者提供当前活动服务的列表。每行一个服务，每行包含一个服务名、协议（例如 tcp）和端口号。

充分利用日志得到服务器信息，对于服务器排错、安全控制等都非常重要。最简单的方法是，通过日志可以发现一些非法连接企图，有关日志属性在表 5-16 中已有比较详细的说明。

3. xinetd 文件的生成

可以使用系统自带的 xinetd 文件修改生成，但是修改后一定要重新使 xinetd 文件有效。下面是两种使 xinetd 有效的方式。

```
#killall -HUP xinetd  （先使用命令killall）
#/usr/sbin/xinetd
```

或

```
#/etc/rc. d/init. d/xinetd restart
```

4. xinetd 配置样本

下面是一个 xinetd 配置的例子。

```
#
#xinetd配置样本
#
defaults
 {
          log_type            =FILE/var/log/servicelog
          log_on_success      =PID
          log_on_failure      =HOST RECORD
          only_from           =128.138.193.0128.138.204.0
          only_from           =128.138.252.1
          instances           =10
          isabled             =rstatd
  }
#
#注意1：protocol属性不是必需的
#注意2：instances属性覆盖了默认设定
#注意3：only_from对于IP地址作了访问控制，只允许
#128.138.193和128.138.204子网及128.138.252.1主机的访问
#
service login
 {
       socket_type            =stream
       protocol               =tcp
       wait                   =no
       user                   =root
       server                 =/usr/etc/in. rlogind
       instances              =UNLIMITED
       }
#
#注意1：instances属性覆盖了默认设定
#注意2:此处的log_on_success是追加方式
#
service shell
  {
        socket_type           =stream
        wait                  =no
        user                  =root
        instances             =UNLIMITED
        server                =/usr/etc/in. rshd
        log_on_success        +=HOST RECORD
 }
```

```
service ftp
 {
      socket_type                 -stream
             wait                 -no
             nice                 -10
            user                  =root
            server                =/usr/etc/in. ftpd
            server_args           =-1
            instances             =4
            log_on_success        +=DURATION HOST USERID
            access_times          =2:00-9:0012:00-24:00
 }

#限制telnet会话使用8MB内存、子进程20 CPU秒
service telnet
   {
     socket_type                  =stream
     wait                         =no
     nice                         =10
     user                         =root
     server                       =/usr/etc/in.telnetd
     rlimit_as                    =8M
     rlimit_cpu                   =20
     }
#
#本条和下一条指定了内部服务,
#因为是使用不同socket类型的同名服务,
#id属性唯一标识每个条目
#
service echo
     {
     id                           =echo-stream
     type                         =INTERNAL
     socket_type                  =stream
     user                         =root
     wait                         =no
     }

service echo
     {
     id                           =echo-dgram
     type                         =INTERNAL
     socket_type                  =dgram
     user                         =root
     wait                         -no
     }

service servers
     {
     type                         -INTERNAL UNLISTED
     protocol                     -tcp
```

```
      port                      =9099
      socket_type               =stream
      wait                      =no
     }

#
#RPC服务样本
#
service rstatd
       {
      type                      =RPC
      socket_type               =dgram
      protocol                  =udp
      server                    =/usr/etc/rpc.rstatd
      wait                      =yes
      user                      =root
      rpc_version               =2-4
      env                       =LD_LIBRARY_PATH=/etc/securelib
      }

#
#未列表服务样本
#
service unlisted
      {
      type                      =UNLISTED
      socket_type               =stream
      protocol                  =tcp
      wait                      =no
      server                    =/home/user/some_server
      port                      =20020
     }
```

5. xinetd 的安全配置实例

```
/etc/xinetd.c/services文件

#
#default: off
#description: An internal xinetd service, listing active services
service services
 {
      disable                   =yes
      type                      =INTERNAL UNLISTED
      port                      =9098
      socket_type               =stream
      protocol                  =tcp
      wait                      =no
      only_from                 =127.0.0.1
}
#
#/etc/xinetd.c/services文件到此结束
----------------------------------------------------------------------
/etc/xinetd.c/servers文件

#
```

```
#default: off
#description: An internal xinetd service, listing active servers service
servers
 {
     disable                    =yes
     type                       =INTERNAL UNLISTED
     port                       =9099
     socket_type                =stream
     protocol                   =tcp
     wait                       =no
     only_from                  =127.0.0.1
}
#
#/etc/xinetd.c/servers文件到此结束
------------------------------------------------------------------------
/etc/xinetd/telnet文件

#
#default: on
#description: The telnet server serves telnet sessions; it uses\
#     unencrypted username/password pairs for authentication
service telnet
 {
     disable                    =no
     flags                      =REUSE
     socket_type                =stream
     wait                       =no
     user                       =root
     server                     =/usr/sbin/in.telnetd
     log_on_failure             +=USERID
}
#
#/etc/xinetd/telnet文件到此结束
------------------------------------------------------------------------
/etc/xinetd/wu-ftpd文件

#
#default: on
#description: The wu-ftpd FTP server serves FTP connections. It uses\
#normal,unencrypted usernames and passwords for authentication
service ftp
 {
     disable                    =no
     socket_type                =stream
     wait                       =no
     user                       =root
     server                     =/usr/sbin/in.ftpd
```

```
      server args                 =-l-a
      log on success              +=DURATION
      nice                        =10
}
#
#/etc/xinetd/wu-ftpd文件到此结束
------------------------------------------------------------------------
/etc/xinetd/ssh文件

#
service ssh
 {
      socket_type=stream
      protocol=tcp
      instances=10
      nice=10
      wait=no
      user=root
      server=/usr/local/sbin/sshd
      server_args=-i
      log_on_failure+=USERID
      only_from=192.168.0.0
      no_access=192.168.44.0
      no_acces s+=192.168.33.0
}
#
#/etc/xinetd.d/ssh文件到此结束
```

5.3.3 Sudo

1. Sudo 简介

Sudo 是一个用来允许系统管理员给予特定的普通用户（或者用户组）有限的超级用户特权，使其能够以超级用户或其他用户的身份运行一些（或者所有）命令并且记录其所有命令和参数的程序。最基本的原则是在普通用户可以完成工作的范围内给予尽可能少的特权。Sudo 是自由软件，以 BSD 风格的许可证发布。

Sudo 以命令的方式操作，它不是 shell 的替代品。Sudo 具有以下特征：

- 限制用户在每台主机上运行的命令。
- Sudo 对于每个命令都进行记录，可以清楚地审核谁做了什么。
- Sudo 为“通行证系统”提供标记日期的文件，例如每 5 分钟更新一次通行证就可以避免合法用户离开终端的时候被他人盗用。
- Sudo 配置文件是 sudoers。同一个 sudoers 可以在多台机器上使用，方便了定义用户特权的灵活性。

如果想得到详细的 Sudo 手册，可以访问：

http://www.courtesan.com/sudo/man/sudoers.html

2．Sudo 的获取和安装

1）首先获取 Sudo

Sudo 的最新版为 2002 年 4 月 25 发布的 1.6.6。

可以通过以下几种方式取得 Sudo。

- ftp 方式：ftp://ftp.sudo.ws:/pub/sudo/。
- web 方式：http://www.sudo.ws/dist/。

以下是一些镜像站点：

http://www.rge.com/pub/admin/sudo/（Rochester，New York，USA）

http://ftp.cerias.purdue.edu/pub/tools/unix/sysutils/sudo/（West afayette，Indiana，USA）

http://core.ring.gr.jp/archives/misc/sudo/（Japan）

http://www.ring. gr. jp/archives/misc/sudo/

- 匿名 cvs 方式：具体参见 http://www.courtesan.com/sudo/anoncvs.htm l。

2）安装 Sudo

对于源码包，进行如下操作：

```
#./configure; make; make install
```

对于 rpm 包，进行如下操作：

```
#rpm -ivh sudo*
```

3．Sudo 的配置实例

```
/etc/sudoers样本文件

#
#该文件必须以root身份使用visudo命令编辑
#
#阅读sudoers手册以得到书写sudoers文件的细节
#

##
#指定别名用户
##
User_Alias FULLTIMERS=millert, mikef, dowdy
User_Alias PARTTIMERS=bostley, jwfox, crawl
User_Alias WEBMASTERS=will, wendy, wim

##
#指定别名Runas
##
Runas Alias OP-root, operator
Runas Alias DB-oracle, sybase
```

```
##
#指定别名主机
##
Host_Alias SPARC=bigtime, eclipse, moet, anchor: \
    SGI=grolsch, dandelion, black:\
    ALPHA=widget, thalamus, foobar:\
    HPPA=boa, nag, python
Host_Alias CUNETS=128.138.0.0/255.255.0.0
Host_Alias CSNETS=128.138.243.0, 128.138.204.0/24, 128.138.242.0
Host_Alias SERVERS=master, mail, www, ns
Host_Alias CDROM=orion, perseus, hercules

##
#指定别名命令
##
Cmnd_Alias DUMPS=/usr/sbin/dump, /usr/sbin/rdump, /usr/sbin/restore, \
           /usr/sbin/rrestore, /usr/bin/mt
Cmnd_Alias KILL=/usr/bin/kill
Cmnd_Alias PRINTING=/usr/sbin/lpc, /usr/bin/lprm
Cmnd_Alias SHUTDOWN=/usr/sbin/shutdown
Cmnd_Alias HALT=/usr/sbin/halt, /usr/sbin/fasthalt
Cmnd_Alias REBOOT=/usr/sbin/reboot, /usr/sbin/fastboot
Cmnd_Alias SHELLS=/usr/bin/sh, /usr/bin/csh, /usr/bin/ksh, \
           /usr/local/bin/tcsh, /usr/bin/rsh, \
           /usr/local/bin/zsh
Cmnd_Alias SU=/usr/bin/su
Cmnd_Alias VIPW=/usr/sbin/vipw, /usr/bin/passwd, /usr/bin/chsh, \
           /usr/bin/chfn

##
#覆盖默认值
##
Defaults             syslog=auth
Defaults: FULLTIMERS  !lecture
Defaults: millert     !authenticate
Defaults@SERVERS      log_year, logfile=/var/log/sudo. log

##
#指定用户
##

#root用户和wheel组的用户可以以任何身份在任何机器上运行任何程序
root  ALL= (ALL) ALL
%wheel  ALL= (ALL) ALL

#专职系统管理员可以在任何机器上运行任何程序而不需要密码
FULLTIMERS ALL-NOPASSWD:ALL

#兼职系统管理员可以运行任何程序，但是需要密码
PARTTIMERS ALL-ALL
```

```
# jack可以在CSNETS的任何机器上运行任何程序
jack  CSNETS=ALL

#lisa可以在CUNETS（一个B类网）运行任何命令
lisa  CUNETS=ALL

# operator可以保留/usr/oper/bin/下的任何文件和命令
operator ALL=DUMPS, KILL, PRINTING, SHUTDOWN, HALT, REBOOT, \
/usr/oper/bin/

#joe只能转换身份为operator
joe  ALL=/usr/bin/su operator

#pete可以更改hp snakes上除了root以外的所有用户的密码
pete  HPPA=/usr/bin/passwd  [A-z] *, !/usr/bin/passwd root

#bob可以指定runas别名列出的任何用户的身份在sparc和sgi机器上运行任何程序
#
bob  SPARC= (OP) ALL:SGI= (OP) ALL

#jim可以运行biglab网络组机器上的任何程序
jim + biglab=ALL
#网络组的用户可以管理打印机，也可以添加和删除用户
+secretaries ALL=PRINTING, /usr/bin/adduser, /usr/bin/rmuser

#fred不提供密码就可以运行oracle或sybase等程序
fred  ALL= (DB) NOPASSWD:ALL

#在alphas上，john可以转换身份为除了root之外的任何人，但是不允许标记
john  ALPHA=/usr/bin/su [!-] *, !/usr/bin/su*root*

#除了SERVERS中的别名主机外，jen可以在任何机器上运行任何程序
jen  ALL, !SERVERS=ALL

#除了SU和SHELLS别名中的命令外，jill可以运行/usr/bin/目录下的任何命令
jill  SERVERS=/usr/bin/, !SU, !SHELLS
#steve可以用户operator身份运行/usr/local/op_commands/目录下的任何命令
steve  CSNETS= (operator) /usr/local/op_commands/

#matt需要权限杀掉已经被挂起的程序
matt  valkyrie=KILL

#WEBMASTERS的别名用户（will、wendy和wim）可以用www
#（拥有主页的）用户身份运行任何命令或仅仅转换身份为www
WEBMASTERS www= (www) ALL, (root) /usr/bin/su www

#任何人不需要密码就可以用mount/unmount别名CDROM指定的任何机器上的CD-ROM
ALL CDROM=NOPASSWD:/sbin/umount/CDROM, \
      /sbin/mount-o nosuid\, nodev/dev/cd0a/CDROM
```

5.3.4 安全检查工具 nessus

1. nessus 简介

nessus 是一个远程安全扫描器。它是自由软件，功能强大，更新极快，易于使用。安全扫描器的功能是对指定网络进行安全检查与弱点分析，确定是否有破坏者闯入或存在某种方式的误用，寻找导致对手攻击的安全漏洞。nessus 的安全检查由 plug-ins 插件完成。例如 useless services 类的 Echo port open 和 Chargen 插件用来测试主机是否容易受到已知的 echo-chargen 攻击；backdoors 类的 pc anywhere 插件用来检查主机是否运行了 BO、pcAnywhere 等后台程序。除插件外，nessus 还提供描述攻击类型的脚本语言（NSSL）来进行附加的安全测试。

2. nessus 的编译安装

（1）获取 nessus，下载地址为 http://www.nessus.orgdownload.html。

为了编译 nessus，需要获得以下 4 个文件，编译过程必须按顺序执行。

```
nessus-libraries-x.x.tar.gz
libnasl-x.x.tar.gz
nessus-core.x.x.tar.gz
nessus-plugins.x.x.tar.gz
```

（2）使用 tar -xzvf*-*解开上述 4 个文件。

（3）安装 nessus 库。

```
$cd nessus-libraries
$./configure
$make
#make install        （以root身份执行）
```

（4）安装 libnasl。

```
$cd libnasl
$./configure
$make
#make install        （以root身份执行）
```

（5）安装 nessus-core。

```
$cd nessus-core
$./configure
$make
#make install        （以root身份执行）
```

（6）安装 nessus-plugins。

```
$cd nessus-plugins
$./configure
$make
#make install        (以root身份执行)
```

注意：确信/etc/ld.so.conf 包含路径/usr/local/lib，执行 ldconfig。

3. nessus 的配置

1）nessus 配置步骤

（1）首先创建账户。

nessusd（守护进程）服务端有自己的用户数据库，每个用户都有一套约束。可以在整个网络内共享一个服务端，而每个管理员有测试自己的网络部分。

```
$nessus-adduser
Addition of a new nessusd user
--------------------------------
Login: renaud              //输入用户名
Password: secret           //输入用户口令
Authentification type (cipher or plaintext) [cipher]: cipher
Now enter the rules for this user, and hit ctrl-D once you are done:
 (the user can have an empty rule set)
^D
Login: renaud
```

（2）配置 nessus daemon。配置文件是/usr/local/etc/nessus/nessusd.conf，可以为 nessusd 设置几个参数。通常建议使用标准文件不做更改。

（3）最后启动 nessusd。

```
#nessusd-D (root身份执行)
```

2）Nessus Setup 对话框

下面讲解 Nessus Setup 对话框中各选项卡的配置。

- 使用客户端。

图 5-2 是 Nessusd host 选项卡，在选项卡中输入 Nessus 服务器所在的 Linux 机器 IP 地址，端口号及加密方式不需要改动。下面输入用户名，单击 Log in 按钮登录。

- 安全配置。

安全配置按图 5-3 所示 Plugins 选项卡配置。

- 喜爱的插件。

如图 5-4 所示的在 Prefs 选项卡中上半部分是插件选择，下面是插件所能检查的攻击方法。单击每个攻击方法会弹出一个对话框介绍它的危害性及解决方法，建议选择全部的插件以增加安全扫描的完整性。

- 扫描选项。

在如图 5-5 所示 Scan options 选项卡中选择扫描选项。

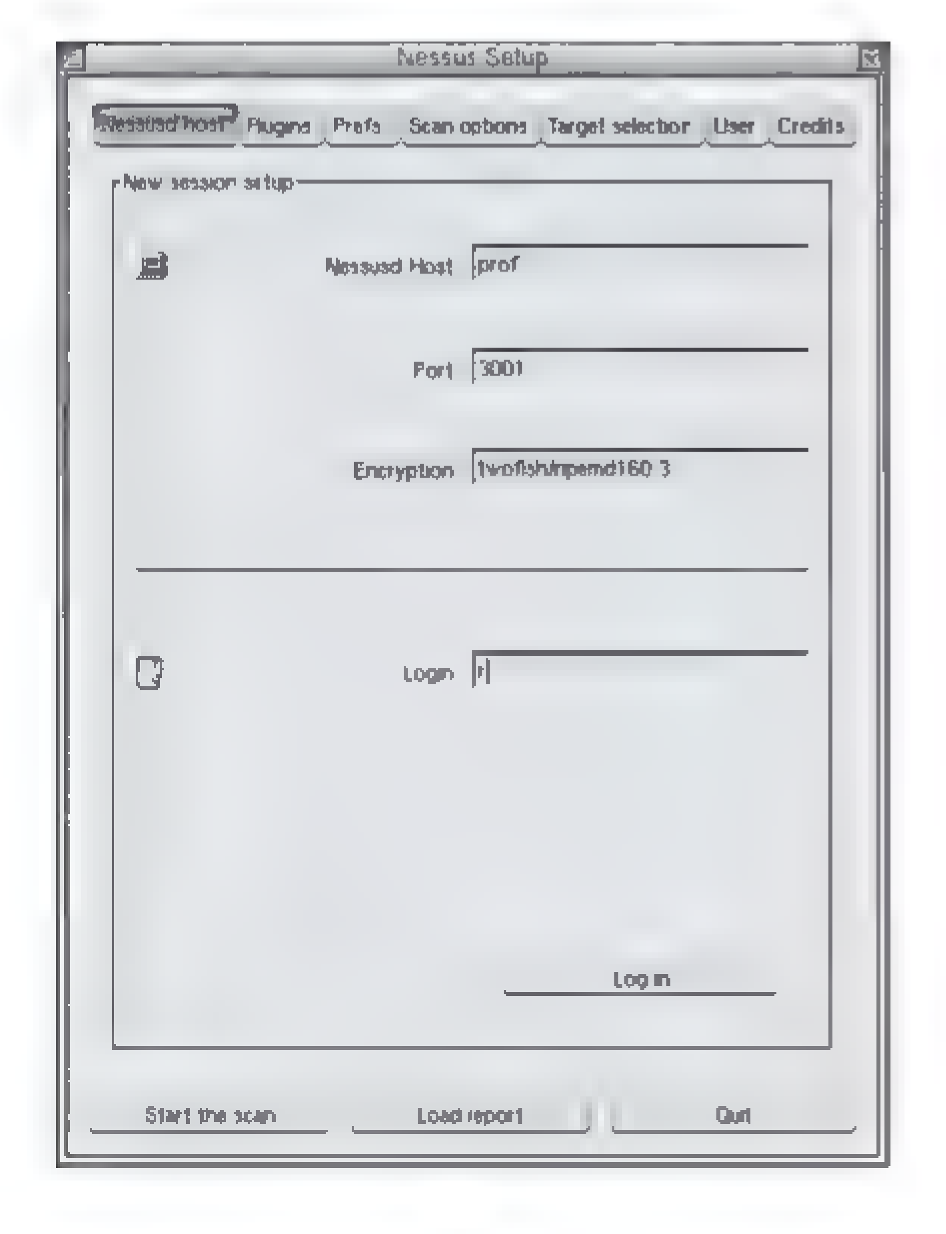


图 5-2

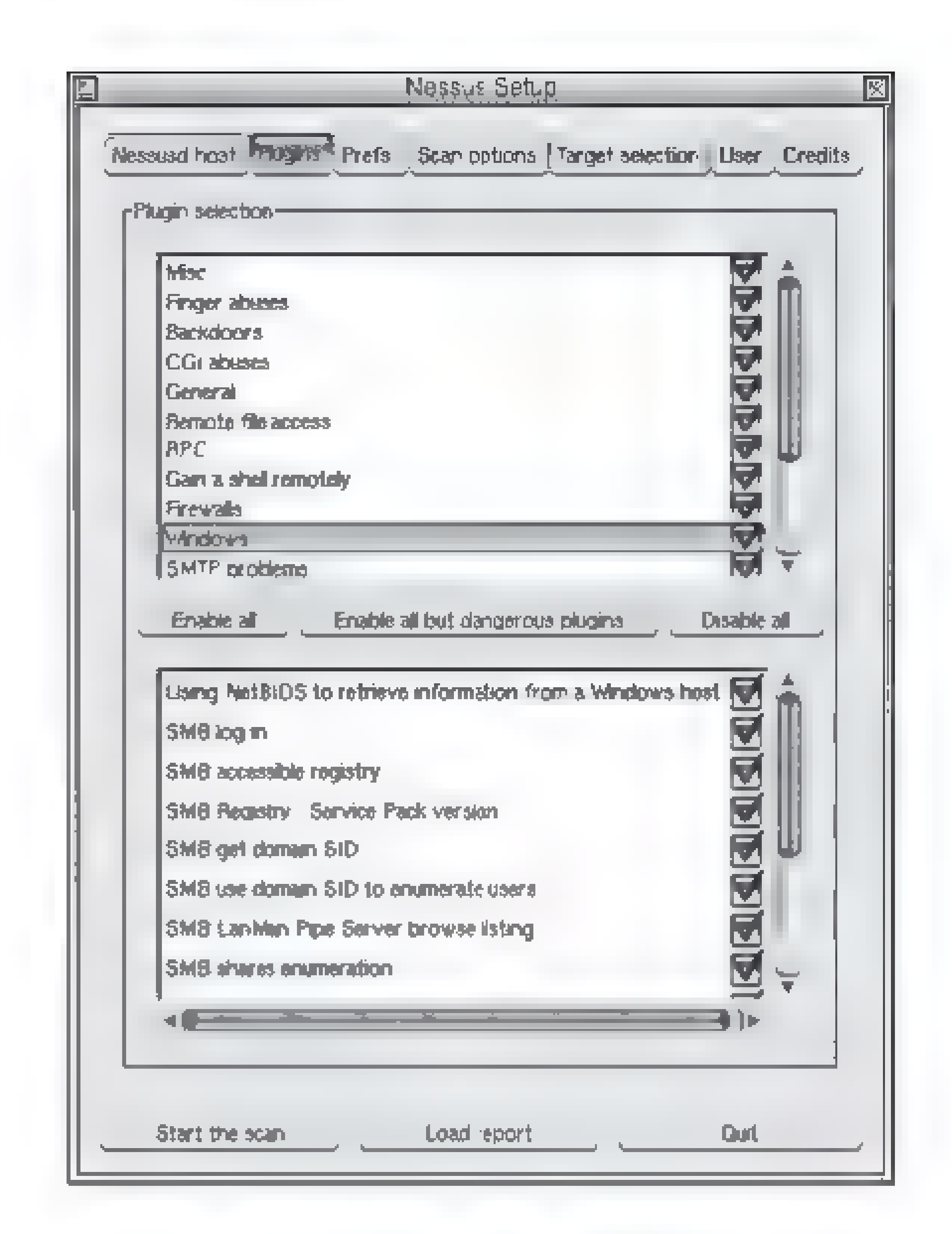


图 5-3

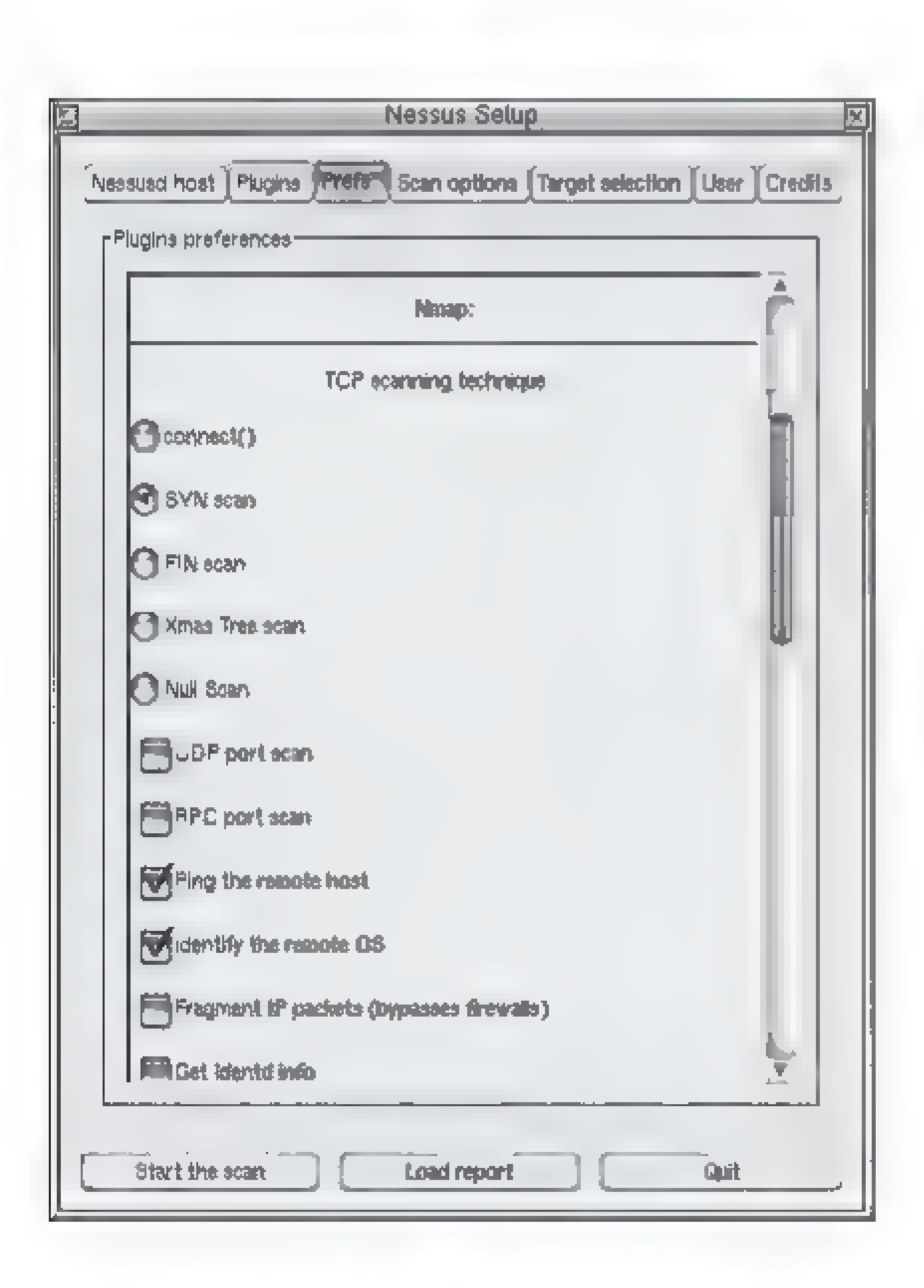


图 5-4

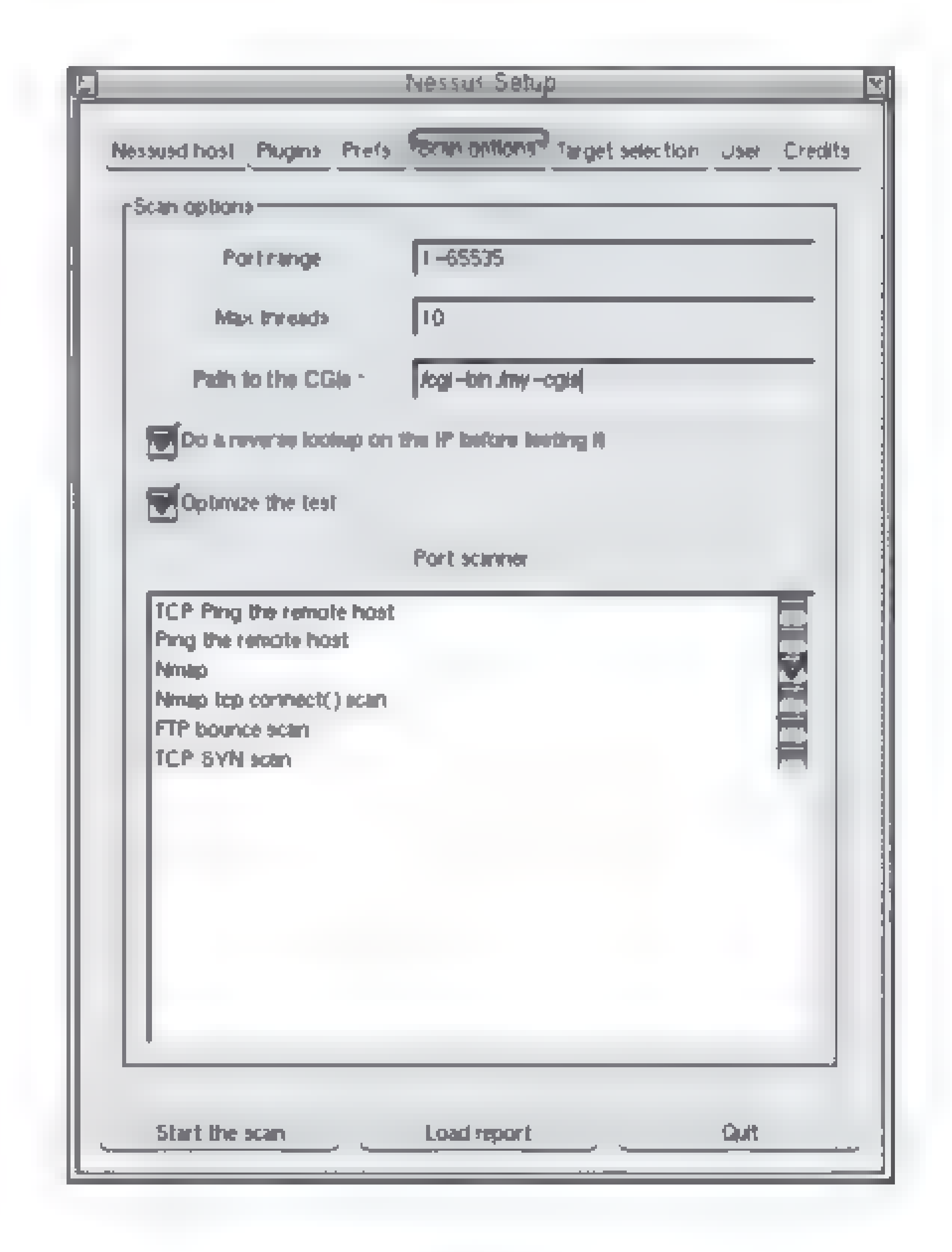


图 5-5

- 目标选择。

在如图 5-6 所示 Target selection 选项卡中进行目标选择。

- 规则选择。

在如图 5-7 所示 User 选项卡中选择规则。

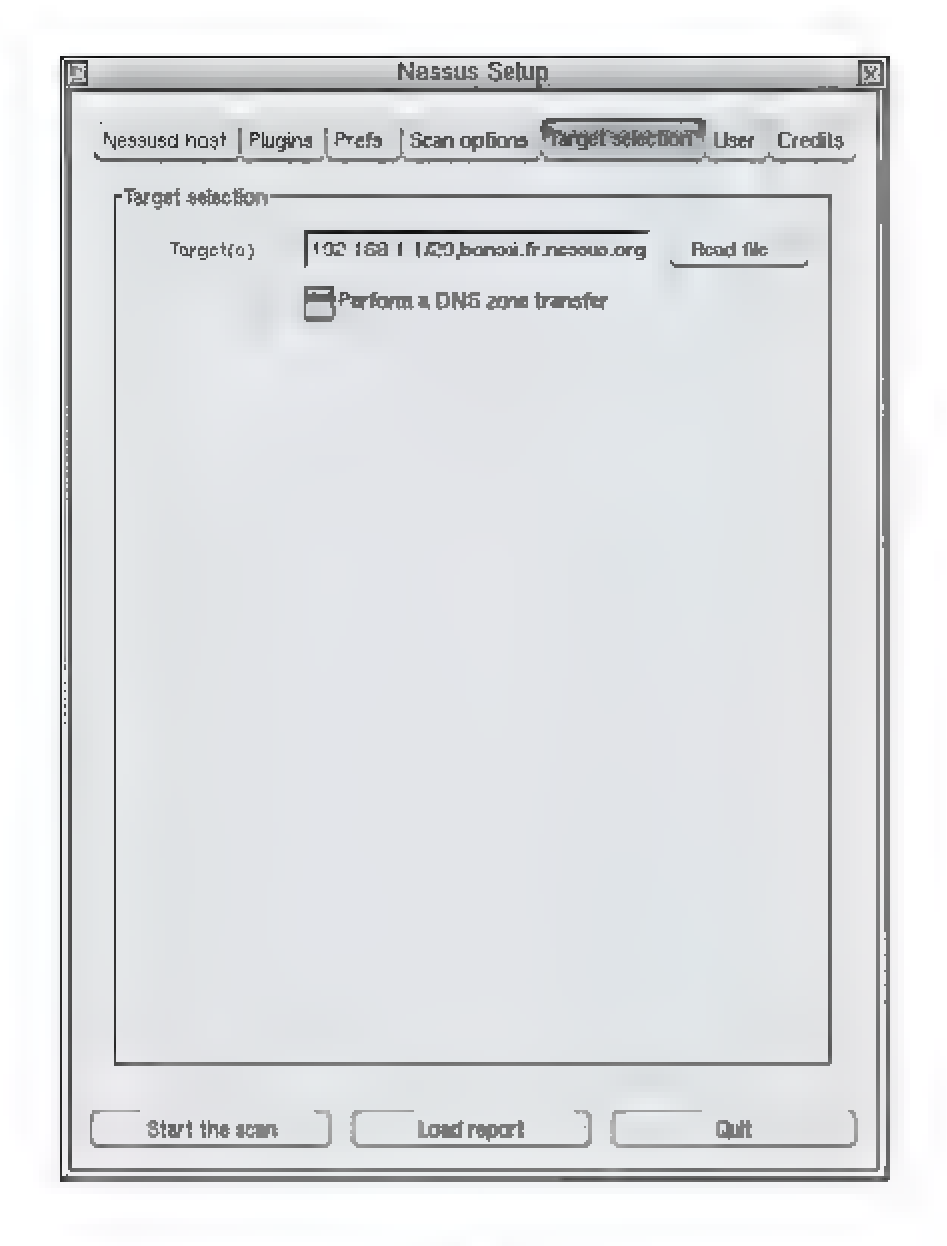


图　5-6

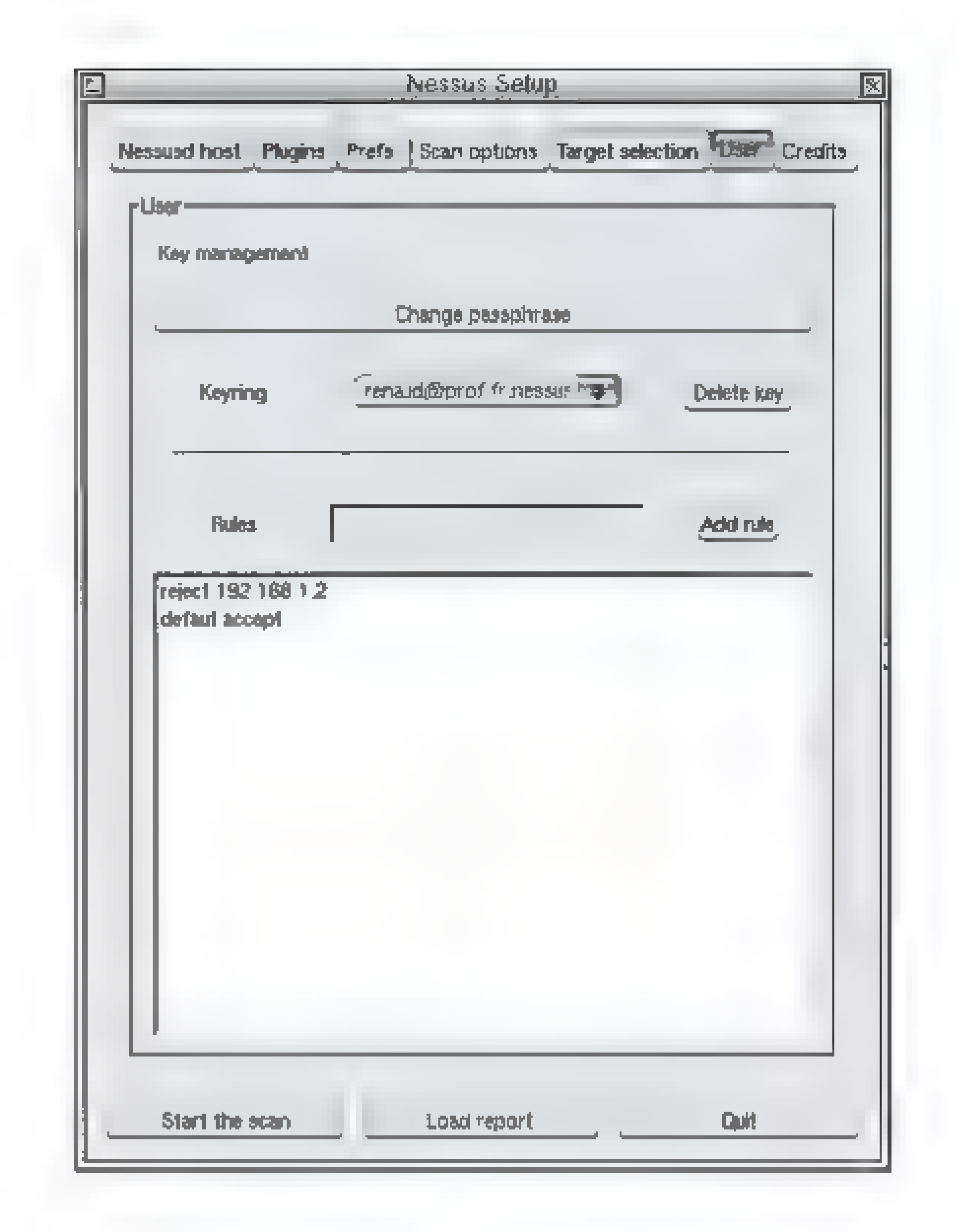


图　5-7

- 开始测试。

单击 Start scan 按钮，开始测试，如图 5-8 所示。

图　5-8

- 测试结果。

如图 5-9 所示，窗口左边列出了所有被扫描的主机，用鼠标单击主机名称，在窗口右边会列出经扫描发现的该主机安全漏洞。单击安全漏洞的小图标会列出该问题的严重等级、问题的产生原因及解决方法。最后，还可以将扫描结果以多种格式存盘，作为参考资料供

以后使用。

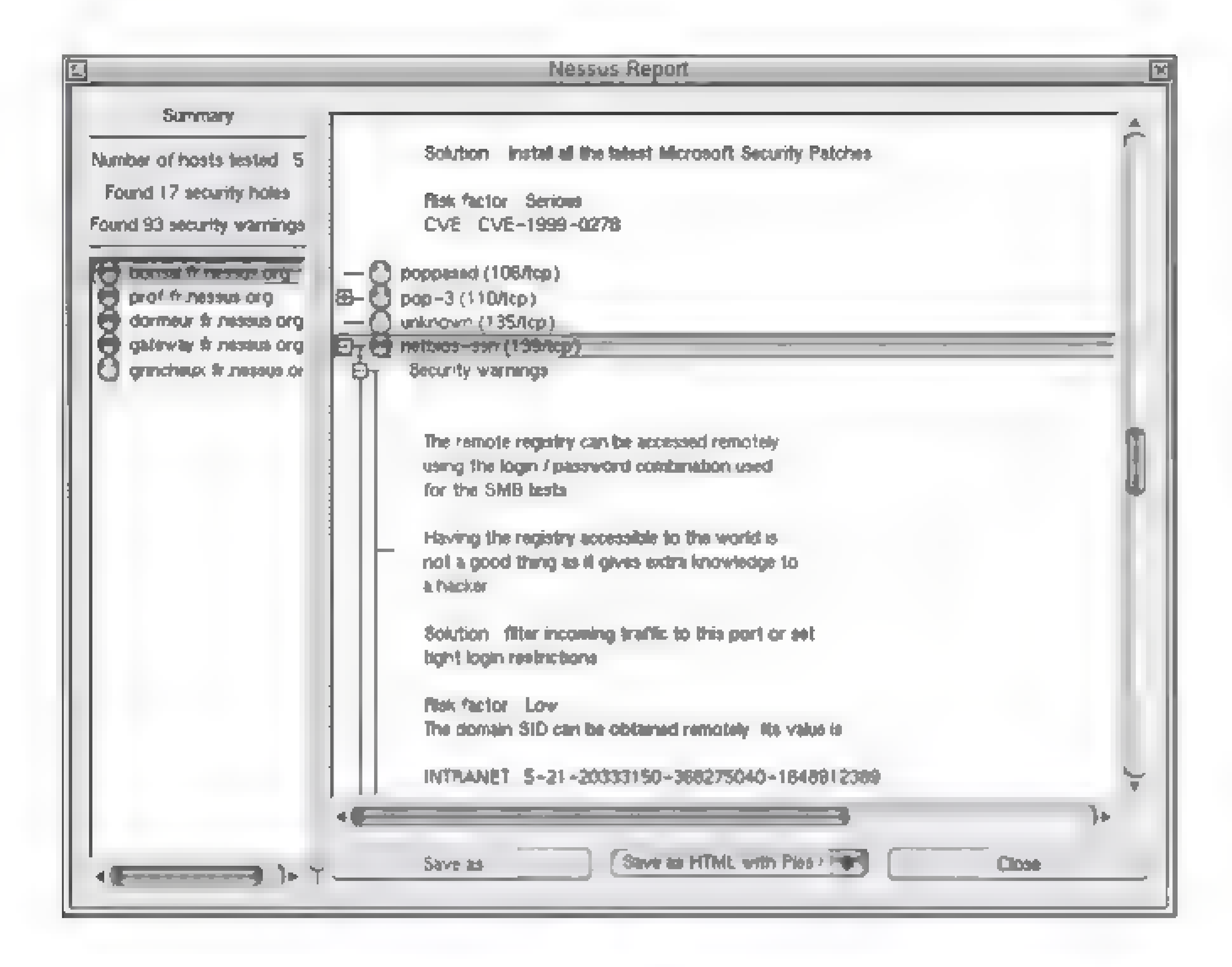


图 5-9

5.3.5 监听工具 sniffit

sniffit 是可以在 Linux、SunOS、Solaris、FreeBSD 和 IRIX 平台运行的网络监听软件，主要用于监听运行 TCP/IP 协议的计算机以监听其不安全性。因为数据包必须经运行 sniffit 的计算机才能进行监听，所以它只能监听同一个网段上的计算机，可以为其增加某些插件以实现额外功能。可以配置 sniffit 在后台运行以检测 TCP/IP 端口上用户的输入输出信息。用户可以选择源、目标地址或地址集合，还可以选择监听的端口、协议和网络接口等。sniffit 会将监听到的数据包内容存放在当前工作目录下，可以直接查看。由于需要将网卡置入混杂模式，所以必须用 root 权限运行。

1. sniffit 的获取

可以从 http://reptile.rug.ac.be/～coder/sniffit/sniffit.html 得到。

2. sniffit 的安装

```
#tar xvfz sniff*.*.tgz  //解开压缩文件
#cd***    //进入文件解开后存放的目录
#./configure
#make
#make clean
```

3. sniffit 的主要参数

主要参数如表 5-18 所示。

表 5-18　snifft 的主要参数表

参　数	描　述
-c<file>	通过脚本运行程序
-F<device>	强制使用网络硬盘
-i	交互模式，可以查看网络中正在连接的机器及其使用的端口号
-n	显示假数据包，包括使用 ARP、RARP 的其他非 IP 数据包
-p<port>	记录连接到<port>的包，port 为 0 记录所有的端口，默认为 0，只用于 TCP 和 UDP 数据包
-s<ip nr/name>	监听从某 IP 发出的数据包，可以使用@通配符选择地址范围，如-s 199.11@
-t<ip nr/name>	监听发送到某 IP 的数据包，可以使用@通配符选择地址范围，如-t 199.11@

注意：-t 或-s 适用于 TCP/UDP 数据包，对 ICMP 和 IP 也进行解释。

4. sniffit 的实例

假定：同一子网的两台主机，L 运行了 sniffer，另一台 Y 的 IP 为 202.206.196.6。

- 检查 sniffer 是否运行且另开一个窗口。

```
sniffit:~/#sniffit -d -p 7 -t 202.206.196.6
```

- sniffer 捕获了 telnet 到对方 7 号端口 echo 服务的包。

```
sniffit:~/$telnet Y7
```

- 截获 Y 上的用户密码。

```
sniffit:~/#sniffit -p 23 -t 202.206.196.6
```

- 截获所有用户通过 Y 接收邮件的 POP3 账号和密码。

```
sniffit:~/#sniffit -p 110 -t 202.206.196.6&
sniffit:~/#sniffit -p 110 -s 202.206.196.6&
```

- 查看 FTP 连接。

```
sniffit:~/#sniffit -p 21 -l 0 -t 202.206.196.6
```

- 截获错误发生的控制信息。

```
sniffit:~/#sniffit -P icmp -b -s 202.206.196.6
```

- 执行脚本。

```
sniffit -c <scriptname>
```

sniffit 监听到的内容通常以 IP 数据文件的形式存储在当前目录下，例如：

```
[root@mail test]#../sniffit -t 202.199.248.11
Supported Network device found.(eth0)
```

```
Sniffit.0.3.7 Beta is up and running....(202.199.248.11)
Gracefull shutdown...
 [root@mail test]#ls
202.112.94.108.34389-                    202.199.248.11.23
202.206.197.227.2080-202.199.248.11.20
```

文件名 202.112.94.108.34389-202.199.248.11.23 表示由 202.112.94.108 的 34389 端口发送到 202.199.248.11 的 23 端口的数据包。

5. ToD 插件

ToD（TOuch of Deatch）是 sniffit 最有名的一个插件，也称“TCP 杀手”。当监听到一个 TCP 连接（包括该连接是某两台主机间的 TCP 连接，与监听程序所在主机无关），可以轻易地将该 TCP 连接切断。这种方法的原理很简单，只要向 TCP 连接中的一台主机发送一个断开连接的 IP 包即可（将 IP 包的 RST 位设置为 1）。

5.3.6 扫描工具 nmap

1. 简介

nmap（Network Mapper）是开放源码的网络探测和安全扫描工具。虽然它主要是用来快速扫描大型网络，但在单主机上也能很好地工作。nmap 可以找到网络上有哪些主机，它们提供了什么服务（端口），运行的是何种操作系统，过滤器/防火墙使用哪些类型的包及其他的许多特征。nmap 可以在绝大多数类型的计算机上运行，有命令行和图形界面版本。

nmap 具有下述特点。

- 灵活：支持多种高级探测技术，包括 UDP、TCP connect、TCP SYN、ftp 代理（bounce 攻击）、反向标志、ICMP、FIN、ACK 扫描、圣诞树（Xmas Tree）、SYN 扫描和空（Null）扫描。可以探测有 IP 过滤、防火墙、路由器和其他限制的网络。
- 强大：nmap 可以扫描拥有万台以上机器的巨型网络。
- 可移植：支持绝大多数操作系统类型，包括 Linux、Open/Free/Net BSD、Solaris、IRIX、Mac OS X、HP-UX 和 Sun OS 等。测试版也提供 Windows 支持。
- 容易：nmap 为高级用户提供大量丰富的高级特征，包括通过 TCP/IP 协议栈特征探测操作系统类型、秘密扫描、动态延时和重传计算、并行扫描、通过并行 ping 扫描探测关闭的主机、诱饵扫描、避开端口过滤检测、直接 RPC 扫描（无需端口映射）、碎片扫描，以及灵活的目标和端口设定。
- 自由：nmap 遵循 GPL 版权，可以免费下载修改发布。
- 良好的文档支持：有丰富的 man 手册、向导等帮助文档。
- 技术支持：除了可以写信给作者之外，还可以加入邮件列表。
- 流行：每天都有成千的人下载 nmap，涵盖各种操作系统（Redhat Linux、Debian Linux、FreeBSD 和 OpenBSD 等），为 nmap 提供了活跃的开发和用户支持社区。

可以从 nmap 的主页 http://www.insecure.org/nmap/index.html 下载 nmap 或者获得更多的信息。

2. nmap 的使用

nmap 的语法格式

nmap [扫描类型] [选项] <主机或网络号 #1… [#N] >

- 扫描类型

扫描类型是可选的，各类型如表 5-19 所示。

表 5-19 nmap 的扫描类型表

扫描类型	描 述
-b<ftp 中转主机>	FTP 反弹攻击（bounce attack）：连接到防火墙后的 FTP 服务器进行端口扫描；-b 选项的参数指定要作为代理的 FTP 服务器，语法格式为-b username：pass-word @server：port
-sA	ACK 扫描：通常用来穿过防火墙的规则集，这有助于确定一个防火墙功能比较完善或是简单的包过滤程序，只是阻塞进入的 SYN 包。这种扫描方式不能找出处于打开状态的端口
-sF -sX -sN	秘密 FIN 数据包扫描、圣诞树（Xmas Tree）扫描和空（Null）扫描模式：在 SYN 扫描无法确定的情况下使用，可以逃过一些防火墙和包过滤软件的干扰
-sP	ping 扫描：检查网络上哪些主机正在运行。nmap 在任何情况下都进行 ping 扫描，只在目标主机处于运行状态时才进行后继扫描，当只检查目标主机是否运行才用该选项
-sR	RPC 扫描：与除诱饵扫描外，其他端口扫描方法结合使用，检查所有处于打开状态的端口是否为 RPC 端口，若是则确定其软件及版本号
-sS	TCP 同步扫描（TCP SYN）：通常称为半开扫描（half-open），用于检查目标端口是否有程序监听，需 root 权限定制 SYN 数据包。很少有系统能够把这种扫描记入系统日志
-sT	TCP connect 扫描：最基本的 TCP 扫描方式，用于检查目标端口是否有程序监听，无需 root 权限即可自由使用。这种扫描会被目标主机的日志记录下大批的连接请求及错误信息
-sU	UDP 扫描：检查某台主机提供哪些 UDP（用户数据报协议，RFC768）服务
-sW	对滑动窗口的扫描：非常类似于 ACK 扫描，但有时可以检测到处于打开状态的端口

- 选项

选项是可选的，其含义如表 5-20 所示。

表 5-20 nmap 的通用选项

参 数	描 述
-P0	扫描前不必 ping 主机，用于有不允许 ICMP echo 请求穿过的防火墙的网络
-PT	扫描前用 TCP ping 确定哪些主机正在运行，使用-PT<端口号>来设定目标端口，默认 80，因为通常不会过滤该端口
-PS	对于 root 用户，令 nmap 使用 SYN 包而不是 ACK 包对目标主机进行扫描
-PI	令 nmap 使用真正的 ping（ICMP echo 请求）扫描目标主机是否正在运行
-PB	默认的 ping 扫描选项，它使用 ACK（-PT）和 ICMP（-PI）两种扫描类型并行扫描
-O	激活对 TCP/IP 指纹特征（fingerprinting）的扫描，获得远程主机的标志，即检测目标主机操作系统网络协议栈的特征来获知目标主机操作系统的类型
-I	打开 nmap 的反向标志扫描功能
-f	令 nmap 使用碎片 IP 数据包发送 SYN、FIN、XMAS 和 NULL。使用碎片数据包增加包过滤、入侵检测系统的难度，使其无法知道你的企图。不过有些程序在处理这些碎片包时会有麻烦，虽然包过滤器和防火墙不能防止这种方法，但有很多网络出于性能上的考虑，禁止数据包的分片
-v	冗余模式，给出扫描过程中的详细信息，使用-d 选项可得到更详细的信息

续表

参　数	描　述
-h	快速参考选项
-oN <logfilename>	将扫描结果重定向到一个可读文件 logfilename 中
-oM <logfilename>	将扫描结果重定向到 logfilename 文件。若使用它来代替 logfilename，输出被重定向到标准输出而覆盖正常的输出。同时使用-v 选项，可在屏幕上打印出其他信息
-oS <logfilename>	将扫描结果重定向到 logfilename 文件，使用它则重定向到标准输出
-resume <logfilename>	恢复中断的扫描项（使用中断前的选项），接着 logfilename 日志文件中的最后一次成功扫描进行新的扫描
-iL <inputfilename>	从 inputfilename 文件读取扫描目标，使用它从标准输入读取主机名
-iR	令 nmap 随机挑选主机扫描
-p <端口范围>	选择要进行扫描的端口号范围，默认为 1～1024，以及 nmap-services 文件中定义的端口列表
-F	快速扫描模式，只扫描 nmap-services 文件中列出的端口
-D <decoy1[，decoy2][，ME]，…>	使用诱饵扫描方法对目标网络/主机进行扫描，可以有效地对付一些积极防御机制，很好地隐藏你的 IP 地址。每个诱饵主机名之间用逗号分开，ME 选项代表自己的主机，与诱饵主机名混杂在一起
-S <IP_Address>	nmap 可能无法确定源址（由 nmap 提示），使用该选项给出你的 IP 地址。在欺骗扫描时使用该选项可令目标认为是其他的主机对自己进行扫描
-e	指定 nmap 发送和接收数据包的接口
-g <portnumber>	设置扫描源端口
-r	禁止 nmap 打乱被扫描端口的顺序
--randomize_hosts	在 nmap 扫描前，打乱每组扫描中的主机顺序，令扫描更不易被网络监视器发现，与--scan_delay <milliseconds>选项组合使用效果更佳
-M <max sockts>	设置进行 TCP connect 扫描时用套接字进行扫描的最大数

- 主机或网络号

在 nmap 的所有参数中，只有目标参数主机或网络号是必须给出的。可以在命令行直接输入一个主机名或者一个 IP 地址。若希望扫描某个 IP 地址的一个子网，可以在主机名或 IP 地址的后面加上“/掩码”，掩码为/0 扫描整个网络，为/32 只扫描该主机，为/24 扫描 C 类地址，为/16 扫描 B 类地址。下面三种形式等价指定 128.210.*.*∶∶128.210.*.*、128.21-.0-255.0-255 或 128.210.0.0/16。

3. 扫描示例

1）Ping 扫描

扫描 192.168.7.0 网络指定端口：

```
#nmap -sP -PT80 192.168.7.0/24
TCP probe port is 80

Starting nmap V.2.12 by Fyodor (fyodor@dhp.com, www.insecure.org/nmap/)
Host (192.168.7.11) appears to be up.
Host (192.168.7.12) appears to be up.
Host (192.168.7.76) appears to be up.
Nmap run completed --256IP addresses (3 hosts up) scanned in1 second
```

2）端口扫描（Port Scanning）

```
#nmap -sT 192.168.7.12
Starting nmap V.2.12 by Fyodor (fyodor@dhp.com, www.insecure.org/nmap/)
Interesting ports on (192.168.7.12) :
Port State Protocol Service
7 open tcp echo
9 open tcp discard
13 open tcp daytime
19 open tcp chargen
21 open tcp ftp
…
Nmap run completed --1 IP address (1host up) scanned in3 seconds
```

3）隐蔽扫描（Stealth Scanning）

```
#nmap -sS 192.168.7.7

Starting nmap V.2.12 by Fyodor (fyodor@ dhp.com, www.insecure.org/nmap/)
Interesting ports on saturnlink.nac.net (192.168.7.7) :
Port State Protocol Service
21 open tcp ftp
25 open tcp smtp
53 open tcp domain
80 open tcp http
…

Nmap run completed --1 IP address (1 host up) scanned in 1 second
```

4）UDP 扫描（UDP Scanning）

```
#nmap -sU 192.168.7.7

WARNING:-sU is now UDP scan --for TCP FIN scan use -sF
Starting nmap V.2.12 by Fyodor (fyodor@ dhp.com, www.insecure.org/nmap/)
Interesting ports on saturnlink.nac.net (192.168.7.7) :
Port State Protocol Service
53 open udp domain
111 open udp sunrpc
123 open udp ntp
137 open udp netbios -ns
138 open udp netbios -dgm
177 open udp xdmcp
1024 open udp unknown

Nmap run completed --1 IP address (1 host up) scanned in 2 seconds
```

5）操作系统识别（OS Fingerprinting）

```
#nmap -sS -O 192.168.7.12

Starting nmap V.2.12 by Fyodor (fyodor@ dhp.com, www.insecure.org/nmap/)
Interesting ports on comet (192.168.7.12) :
Port State Protocol Service
7 open tcp echo
9 open tcp discard
13 open tcp daytime
19 open tcp chargen
21 open tcp ftp
…
TCP Sequence Prediction:Class=random positive increments
Difficulty=17818 (Worthy challenge)
Remote operating system guess:Solaris2.6 -2.7
Nmap run completed --1 IP address (1 host up) scanned in 5 seconds
```

6）Ident 扫描（Ident Scanning）

```
#nmap -sT -p 80 -I -O www.yourserver.com

Starting nmap V.2.12by Fyodor (fyodor@ dhp.com, www.insecure.org/nmap/)
Interesting ports on www.yourserver.com (xxx.xxx.xxx.xxx) :
Port State Protocol Service Owner
80 open tcp http root

TCP Sequence Prediction:Class=random positive increments
Difficulty=1140492 (Good luck!)
Remote operating system guess:Linux2.1.122 -2.1.132;2.2.0 -pre1 -2.2.2

Nmap run completed --1 IP address (1 host up) scanned in 1 second
```

7）使用 nmap 监视自己的站点

系统和网络管理员将能发现潜在入侵者对你的系统的探测。

5.3.7 其他安全工具

下面列出了一些常用的安全工具。

```
Netcat              http://www.l0pht.com/~weld/netcat/
Tcpdump             http://www.tcpdump.org/
Snort               http://www.snort.org/
Saint               http://www.wwdsi.com/saint/
Ethereal            http://ethereal.zing.org/
Whisker             http://www.wiretrip.net/rfp/p/doc.asp?id=21&iface=2
Abacus Portsentry   http://www.psionic.com/abacus/portsentry/
DSniff              http://naughty.monkey.org/~dugsong/dsniff/
Tripwire            http://www.tripwire.com/
```

```
Cybercop Scanner    http://www.pgp.com/asp_set/products/tns/ccscanner_
                            intro.asp
Hping2              http://www.kyuzz.org/antirez/hping/
SARA                http://www-arc.com/sara/
```

5.4　配置安全可靠的系统

现在来讨论如何配置一个安全可靠的 Linux 系统，当然，真正的安全是不存在的，我们要做的，是尽量让系统在一开始就避免出现前面说的大部分问题，为此，需要正确配置系统中的大部分服务程序。特别是需要用可靠的服务代替那些容易遭受攻击的服务。

5.4.1　SSH 实践

Telnet 和 FTP 服务是最容易遭到窃听的，因此，通常都要用 SSH 和 sftp 来取代它们。当然，FTP 服务很多时候不能取代，所以很多情况下，需要同时使用 sftp 和 FTP，后者只用来提供匿名 FTP 服务。

SSH 有两个版本：ssh1 和 ssh2，ssh1 和 ssh2 命令基本上一致，只是算法的区别。因为本文的实践环境为 redhat 8.0，而其自带 ssh1，所以只讨论 ssh1。

1．使用前的准备

首先必须检查系统是否自带了 SSH，这可以通过命令 ssh -l 进行测试，如果出现 SSH 的使用提示表示已带 SSH。另外，也要检查是否已有 sshd。若具备了上述条件，则不必手工安装；若不具备，那么需要下载相应软件安装。Openssh 的下载地址为 http://www.openssh.org/。

2．OpenSSH 的安装

（1）如果下载的是 rpm 包，用：

```
rpm -vi opens sh*.rpm
```

（2）如果下载的是源码，用：

```
tar zxvf openssh*.tar.gz
./configure
make
make install
```

（3）安装完成后的测试：

```
ssh [-l login_name] hostname|user@ hostname [command]
```

若 OpenSSH 工作正常，将显示提示信息。例如：

```
[wly@ cs cs]$ssh -l wly localhost
The authenticity of host'localhost (127.0.0.1)'can't be established.
RSA key fingerprint is57:1f:b6:12:72:d6:01:52:9a:c2:83:b3:08:fd:8e:eb.
Are you sure you want to continue connecting (yes/no)?
```

如果是第一次登录到某一台主机，OpenSSH 会出现上面的提示，表示在它的默认标志库中找不到这台机器的内容。输入 yes，这时把主机的“识别标记”加入~/.ssh/know_ hosts 文件。若是第二次访问就不再显示上面的提示信息了。随后，SSH 提示输入账号口令。口令输入完毕则建立了 SSH 连接。以后使用 SSH 类似于 telnet。

3. SSH 密钥的生成和管理

SSH 的密钥是用 ssh-keygen 程序管理的。

SSH 程序的主要语法格式

```
ssh -keygen [-q][-b 密钥位数] -t type [-N 新密钥][-C 注释][-f 输出密钥文件]
ssh -keygen -p [-P 旧密钥][-N 新密钥][-f 密钥文件]
ssh -keygen -e [-f]
ssh -keygen -y [-f 输入密钥文件]
ssh -keygen -c [-P 密钥][-C 注释][-f 密钥文件]
ssh -keygen -l [-f 输入密钥文件]
ssh -keygen -B [-f 输入密钥文件]
ssh -keygen -D reader
ssh -keygen -U reader [-f 输入密钥文件]
```

这里的命令行参数的含义如表 5-21 所示。

表 5-21　命令行参数描述

参　数	描　述
-c	请求变更公有私有密钥文件注释
-D	下载存储在 reader 智能卡的 RSA 公有密钥
-e	读私有或公有 OpenSSH 密钥文件以 SECSH 公有密钥文件形式标准输出
-f	指定密钥文件名
-l	显示指定公有密钥文件指纹
-N	提供新密钥
-p	请求变更私有密钥文件的密钥而不是建立新的私有密钥
-q	用于生成新密钥时停止 ssh-keygen
-t	SSH 可以为 ssh1 生成 RSA 密钥，为 ssh2 生成 RSA 或 DSA 密钥，密钥的类型由-t 指定，对于 ssh1 可能值为 rsa1，对于 ssh2 可能值为 rsa 或 dsa
-U	上载已存在的 RSA 私有密钥到 reader 智能卡
-y	读私有 OpenSSH 格式的文件标准输出 OpenSSH 公有密钥

下面是 ssh-keygen 密钥生成的一个实例。

- [wly@ cs cs]$ssh-keygen -t rsa1

```
Generating public/privater sa1key pair.
Enter file in which to save the key (/home/wly/.ssh/identity):(
```

```
Enter passphrase (empty for no passphrase) : (
Enter same passphrase again: (
Your identification has been saved in/home/wly/.ssh/identity.
Your public key has been saved in/home/wly/.ssh/identity.pub.
The key fingerprint is:
9c:3a:a0:11:9a:39:c2:b5:5c:a6:af:e6:94:df:db:95wly@ cs
 [wly@ cs cs] $
```

- [wly@ cs cs] $ssh-keygen -d

```
Generating public/private dsa key pair.
Enter file in which to save the key (/home/wly/.ssh/id_dsa) :
Enter passphrase (empty for no passphrase) :
Enter same passphrase again:
Your identification has been saved in/home/wly/.ssh/id_dsa.
Your public key has been saved in/home/wly/.ssh/id_dsa.pub.
The key fingerprint is:
fd:93:bc:49:18:bb:88:31:0a:3d:f7:ee:1e:e3:42:99wly@ cs
 [wly@ cs cs] $
```

ssh-keygen -d 与 ssh-keygen -t 类似，默认情况下，只是将一对密钥存为/home/ [user] /.ssh/id_dsa（私有密钥）和/home/ [user] /.ssh/id_dsa.pub（公有密钥）。

公有密钥需要分发到所有想用 SSH 登录的远程主机上去，私有密钥需要好好保管防止他人知道，如：

```
 [wly@ cs cs] $ls -l~/.ssh/identity
-rw-------  1 wly wly   524 10月  2 20:48/home/wly/.ssh/identity
 [wly@ cs cs] $ls -l~/.ssh/id_dsa
-rw-------  1 wly  wly   668 10月  2 21:17/home/wly/.ssh/id_dsa
```

注意这里 ls 的结果显示文件的访问权限必须是-rw------。

如果密钥已经被别人知道，马上生成一对新的密钥。当然，也需要重新分发一次公有密钥。

公有密钥需要经过分发才能使用，为此，需要执行：

```
 [wly@ cs cs] $md.ssh
 [wly@ cs cs] $cp identity.pub authorized_keys
 [wly@ cs cs] $chmod 644.ssh/authorized_keys
```

上述过程需要分别在每个用 SSH 连接的远程服务器上完成。为了对于 authorized keys 保证他人没有写的权限，保证 SSH 工作，chmod 是必须使用的命令。

如果想从不同的计算机登录到远程主机，authorized keys 文件也可以有多个公用密钥。这种情况下必须在新的计算机上重新生成一对密钥，然后重复上述过程。需要注意的是，当取消了主机上的账号之后，别忘了删掉这对密钥。

4. SSH 配置

1）配置 SSH 客户端

OpenSSH 的配置数据可以有三种语法形式，按照优先权从大到小的顺序分别是：命令

行选项、用户配置文件（～/.ssh/config）和系统配置文件（/etc/ssh/ssh_config）。所有的命令行选项均能在配置文件中设置。因为任何配置值都是首次设置时有效，所以指定主机的声明应该位于配置文件的最初，而默认值则放于文件末尾。

下面是/etc/ssh/ssh_config 文件的内容，用户配置文件可以从系统配置文件修改得到，文件中选项的说明如表 5-22 所示。

表 5-22 config 文件说明表

选　项	描　述
BatchMode	如果设为 yes，禁用密钥/密码查询，默认为 no
BindAddress	指定从多接口或 IP 别名的计算机传输所用接口
CheckHostIP	若设为 yes，SSH 将附加检查已知主机文件中的主机 IP 地址，该选项可以进行 IP 地址检查以防止 DNS 欺骗
Cipher	指定 ssh1 所用加密算法
Compression	指定是否使用压缩
CompressionLevel	压缩的级别，1（最快）～9（压缩率最高），默认值为 6
FallBackToRsh	如果远程主机拒绝 ssh 连接（比如没有 sshd 服务），是否自动改用 rsh 连接
ForwardAgent	指定到认证机构的连接是否转发到远程主机
ForwardX11	指定是否 X11 连接自动通过安全通道重定向和 DISPLAY 设置，本地运行远程 X 程序必须设置该选项
GatewayPorts	指定远程主机是否允许连接到本地转发端口
Host	在遇到下一个关键字前，限制匹配关键字后给定模式之一的主机声明，支持通配符“*”和“?”
HostName	指定登录的真实主机名
IdentityFile	指定用户认证的 RSA 或 DSA 识别需要读取的文件，默认为$HOME/.ssh/identity（ssh1）、$HOME/.ssh/id_rsa 和$HOME/.ssh/id_dsa（ssh2）
PasswordAuthentication	指定是否使用密码认证
Port	指定连接到远程主机的端口号，默认值为 22
Protocol	指定 SSH 以优先序应该支持的协议版本
RSAAuthentication	指定是否使用 RSA 认证
User	指定登录用户

```
/etc/ssh/ssh_config文件

# Host *
# ForwardAgent no
# ForwardX11 no
# RhostsAuthentication yes
# RhostsRSAAuthentication yes
# RSAAuthentication yes
# PasswordAuthentication yes
# FallBackToRsh no
# UseRsh no
# BatchMode no
# CheckHostIP yes
# StrictHostKeyChecking ask
# IdentityFile ~/.ssh/identity
# IdentityFile ~/.ssh/id rsa
# IdentityFile ~/.ssh/id dsa
```

```
# Port 22
# Protocol 2, 1
# Cipher 3des
# Ciphers aes128-cbc, 3des-cbc, blowfish-cbc, cast128-cbc, arcfour, aes
  192-cbc, aes2
# EscapeChar ~
Host *
ForwardX11 yes
ForwardAgent no
FallBackToRsh no
#/etc/ssh/ssh_config文件到此结束
```

2）配置 SSH 服务端

SSH 服务器配置文件是/etc/ssh/sshd_config，对于 SSH1.x 和 2.x，OpenSSH 的配置文件是一样的。下面是/etc/ssh/sshd_config 的内容。

```
/etc/ssh/sshd_config文件

#Port 22
#Protocol 2, 1
#ListenAddress 0.0.0.0
#ListenAddress ::

#ssh1的HostKey
#HostKey/etc/ssh/ssh_host_key

#记录

SyslogFacility AUTHPRIV
LogLevel INFO

#认证

PermitRootLogin yes
#是否允许超级用户登录，和telnet不同，SSH默认允许超级用户登录
StrictModes yes

RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile  .ssh/authorized_keys

#禁用rhosts认证
RhostsAuthentication no
#禁读用户的~/.rhosts和~/.shosts文件
IgnoreRhosts yes
#/etc/ssh/ssh_known_hosts中需要host keys
RhostsRSAAuthentication no
IgnoreUserKnownHosts no
#把这个选项设置为no，只允许用户用基于密钥而非基于口令方式登录，
#能在很大程度上提高系统的安全性
PasswordAuthentication yes
```

```
PermitEmptyPasswords no

#X11Forwarding no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PrintMotd yes
#PrintLastLog yes
#KeepAlive yes
#UseLogin no

#MaxStartups10
#no default banner path
#Banner/some/path
#VerifyReverseMapping no

#override default of no subsystems
Subsystem sftp /usr/libexec/openssh/sftp-server
#/etc/ssh/sshd_config文件到此结束
```

5. SSH 实现 telnet

最容易受到监听工具威胁的程序之一是 telnet，一个 sniffer 程序可以轻易地得到你的登录名和密码。解决的方法就是用 SSH 替代 telnet。从使用上讲，SSH 和 telnet 没有不同之处，但是 SSH 将传输中的所有信息加密，确保了传输信息不被窃听。

下面是第一次登录的情况：

```
[wly@ cs cs] $ssh localhost
The authenticity of host'localhost (127.0.0.1) 'can't be established.
RSA key fingerprint is 4b:91:0a:85:7a:ab:f6:1a:f5:51:07:33:4d:ba:ec:e3.
Are you sure you want to continue connecting (yes/no) ?yes
Warning:Permanently added'localhost' (RSA) to the list of known hosts.
wly @localhost's password:
Last login:Wed Oct 2 06:53:42 2002 from 202.206.196.221
[wly@ cs cs] $
```

以后的登录情况中：

```
[wly@ cs cs] $ssh localhost
wly@ localhost's password:
Last login:Wed Oct 2 09:43:16 2002 from cs
[wly@ cs wly] $
```

6. SSH 实现 FTP

与 telnet 类似，FTP 也非常容易受到 sniffer 程序的威胁，与 SSH 可以替代 telnet 相似，

sshd 提供 sftp 服务来加密传输信息，保护信息安全。sftp 程序可以完全替代 FTP 的功能，用法也与之类似。

例如第一次使用会话：

```
[wly@ cs cs]$sftp 202.206.196.220
Connecting to 202.206.196.220…
The authenticity of host'202.206.196.220 (202.206.196.220) 'can't be estab-
lished.
RSA key fingerprint is 4b:91:0a:85:7a:ab:f6:1a:f5:51:07:33:4d:ba:ec:e3.
Are you sure you want to continue connecting (yes/no) ?yes
Warning:Permanently added'202.206.196.220' (RSA) to the list of known hosts.
wly@ 202.206.196.220's password:
sftp>help
Available commands:
cd path      Change remote directory to'path'
lcd path     Change local directory to'path'
chgrp grp path   Change group of file'path'to'grp'
chown own path  Change owner of file'path'to'own'
help          Display this help text
get remote-path [local-path] Download file
…
```

以后的会话：

```
[wly@ cs cs]$sftp 202.206.196.220
Connecting to 202.206.196.220…
wly@ 202.206.196.220's password:
sftp>
```

7. 用 SSH 设置“加密通道”

通过“端口转发”可以实现 SSH 的本地未用端口与远程服务器上运行的某个服务端口间的“加密通道”。当远程服务器运行 SSH 服务器软件时，只要连接到本地端口，所有对本地端口的请求都被 SSH 加密并且转发到远程服务器端口。

1）检查远程服务器是否运行 SSH 服务

```
[wly@ cs cs]$telnet 202.206.196.221 22
Trying 202.206.196.221…
telnet:connect to address 202.206.196.221:Connection refused [wly@ cs cs]$
```

如果出现下面的画面，则说明运行 SSH 服务。

```
[wly@ cs cs]$telnet cs22
Trying 127.0.0.1…
Connected to cs.
Escape character is'^]'.
```

端口转发使用的语法：

```
ssh -f [username@remote host] -L [localport][full name of remote host] :
[remote port][some command]
```

- -f：用于告诉SSH在后台的登录，而不产生提示信息。
- username：用户名。
- some command：用于让SSH执行连接前首先执行一些命令。例如 ssh -L 8888 202.112.26.39:23 把本机的8888端口接收到的信息转发给202.112.26.39的23端口。

可以在~/.ssh/config文件中用LocalForward设置经常使用的一些转发端口。

2）POP的“加密通道”

为POP加上“加密通道”同样可以防止其密码被sniffer程序窃取。另外，SSH的压缩方式可以让邮件传输得更快。

```
ssh -f -C username@ address -L portnumber address:110 sleep 5
```

3）X的“加密通道”

SSH还可以对X数据传输进行加密，若要在本机运行远程SSH服务器上的X程序，需在远程计算机上创建一个~/.ssh/environment文件并添加一行：

```
XAUTHORITY=/home/ [remote user name] /.Xauthority
```

注意：*若用户主目录下不存在.Xauthority，则SSH登录时自动创建。*

例：启动一个X程序（xterm）。

```
ssh -f -X -l [remote user name][remote machine] xterm
[wly@ cs.ssh] $ssh -f -X -l wly cs xterm
The authenticity of host'cs (127.0.0.1) 'can't be established.
RSA key fingerprint is4b:91:0a:85:7a:ab:f6:1a:f5:51:07:33:4d:ba:ec:e3.
Are you sure you want to continue connecting (yes/no) ?yes
Warning:Permanently added'cs' (RSA) to the list of known hosts.
wly@ cs's password:
```

若非首次使用会话，则提示信息为：

```
[wly@ cs.ssh] $ssh -f -X -l wly cs xterm
wly@ cs's password:
```

4）Linuxconf的“加密通道”

Linuxconf是支持远程管理的Linux配置工具。

使用方式：

```
remadmin --exec ssh -l [account] linuxconf --guiproto
```

5）Webmin的“加密通道”

Webmin是一个新的基于浏览器的配置工具。运行在1000端口。加密方式为：

```
ssh -f -l [remote user name][remote host] -L local port: [remote host] :10000
tail -f/etc/motd
```

5.4.2　SSL 实践

下一个任务是用 SSL(https)替换标准的 HTTP 服务，现在很多的 Linux 发行版本已经包含了 apache 2.0，这个版本内置了 SSL 模块。不过，由于很多系统仍然在使用 apache 1.3.x，因此，仍然可能需要自己下载和编译 apache+mod_ssl。

1. 需要的软件

1）Apache Web Server

下载地址：http://www.apache.org/dist/

或者从镜像站点下载：http://www.apache.org/dyn/closer.cgi

2）mod_ssl

下载地址：http://www.modssl.org

3）Open SSL

下载地址：http://www.openssl.org

2. 安装

1）取得并展开软件

```
$lynx http://httpd.apache.org/dist/httpd/apache_1.3.26.tar.gz
$lynx ftp://ftp.modssl.org/source/mod_ssl-2.8.10-1.3.26.tar.gz
$lynx ftp://ftp.openssl.org/source/openssl-0.9.6d.tar.gz
$gzip -d -c apache_1.3.26.t ar.gz|tar xvf-
$gzip -d -c mod_ssl-2.8.10-1.3.26.tar.gz|tar xvf-
$gzip -d -c openssl-0.9.6d.tar.gz|tar xvf-
```

2）编译 OpenSSL

```
$cd openssl-0.9.6d
$./config
$make
$cd..
```

3）编译安装 SSL-aware Apache

```
$cd mod_ssl-2.8.10-1.3.26
$./configure\
   --with-apache=../apache_1.3.26\
   --with-ssl=../openssl-0.9.6d\
   --prefix-/usr/local/apache
$cd..
```

```
$cd apache 1.3.26
$make
$make certificate
$make install
```

4）清理

```
$rm -rf apache_1.3.26
$rm -rf mod_ssl-2.8.10-1.3.26
$rm -rf openssl-0.9.6d
```

5）测试 SSL-aware Apache

用步骤3）中在 make certificate 填写的完全认证域名（FQDN）替代 local-host-name。

```
$/usr/local/apache/bin/httpd-DSSL
$netscape https://local-host-name/
```

6）启动与停止 apache

```
$/usr/local/apache/bin/apachectl startssl
$/usr/local/apache/bin/apachectl stop
```

由于 redhat 8.0 发行套件自带 SSL，所以不必重新编译即可获得 SSL 支持。apachectl 在/usr/sbin 下，httpd.conf 在/etc/httpd/conf/下。

7）测试

```
[wly@ cs sbin]#./apachectl startssl
[wly@ cs sbin]#
[wly@ cs sbin]#./apachectl stop
[wly@ cs sbin]#
[wly@ cs sbin]#lynx https://localhost
```

3. 配置

要使 SSL 生效，需要通过编辑修改/usr/local/apache/etc/httpd.conf 文件来完成，这可以在 httpd.conf 中添加如下的语句：

```
#建立只使用SSLv2协议和密码的SSL服务器
SSLProtocol -all+SSLv2
SSLCipherSuite SSLv2:+HIGH:+MEDIUM:+LOW:+EXP

#下面仅仅授权最强的7位密码
SSLProtocol all
SSLCipherSuite HIGH:MEDIUM

SSLCipherSuite ALL:!ADH:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
<Directory/usr/local/apache/htdocs>

# 通过SGC工具更新，最终拒绝所有未更新的浏览器
```

```
SSLRequire%{SSL CIPHER USEKEYSIZE}>=128
</Directory>

# 除https://hostname/strong/area/及其下属需要强壮的密码外，普遍不限
SSLCipherSuite ALL:!ADH:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL
<Location/strong/area>
SSLCipherSuiteHIGH:MEDIUM
</Location>

# 要求我们的CA认证直接签名的客户端证书
SSLVerifyClient require
SSLVerifyDepth1
SSLCACertificateFile conf/ssl.crt/ca.crt

# 区别对待不同的客户端，假定有两个不同子网，
# 一个的网络地址为192.160.1.0/24，另一个subarea的URL为/subarea
SSLVerifyClient              none
<Directory/usr/local/apache/htdocs/secure/area>
SSLVerifyClient              require
SSLVerifyDepth               5
SSLCACertificateFile conf/ssl.crt/ca.crt
SSLCACertificatePath conf/ssl.crt
SSLOptions                   +FakeBasicAuth
SSLRequireSSL
AuthName                     "Snake Oil Authentication"
AuthType                     Basic
AuthUserFile                 /usr/local/apache/conf/httpd.passwd
require                      valid-user
</Directory>
SSLCACertificateFileconf/ssl.crt/company-ca.crt

<Directory/usr/local/apache/htdocs>
# subarea外部只许可Intranet访问
Order                      deny, allow
Deny                       fromall
Allow                      from192.168.1.0/24
</Directory>

<Directory/usr/local/apache/htdocs/subarea>
# subarea内部许可任何Intranet访问，但是来自因特网的只能是
# HTTPS+Strong-Cipher+Password
# 或者HTTPS+Strong-Cipher+Client-Certificate

# 若使用HTTPS，确信使用了强壮的密码
# 添加真实客户替代首要认证机构
SSLVerifyClient               optional
SSLVerifyDepth                1
SSLOptions                    +FakeBasicAuth+StrictRequire
SSLRequire                    %{SSL_CIPHER_USEKEYSIZE}>=128

# 强迫来自Internet的客户使用HTTPS
```

```
RewriteEngine                on
RewriteCond                  %{REMOTE ADDR}!^192\.168\.1\.[0-9]+$
RewriteCond                  %{HTTPS}!=on
RewriteRule                  .*-[F]

# 许可网络访问和/或首要认证机构
Satisfy                      any

# 网络访问控制
Order                        deny, allow
Deny                         from all
Allow                    192.168.1.0/24

# HTTP首要认证机构
AuthType      basic
AuthName      "Protected Intranet Area"
AuthUserFile  conf/protected.passwd
Require       valid-user
</Directory>
```

5.4.3 构造 chroot 的 DNS

在前面理论篇的存储空间部分，介绍了 chroot，我们知道，DNS 的 bind 服务程序很容易被溢出并且导致致命的漏洞。解决方法一方面是注意更新和升级 bind 服务器程序，另一方面就是要设置 bind，使在被溢出的情况下也不会导致攻击者获得 shell。后一种方法可以用 Lids 来实现，也可以通过 chroot 做到。这里就谈谈如何利用 chroot-bind 构建 DNS 服务器，进一步加强 bind 的安全性。

1．安装

1）安装 bind

从 http://www.isc.org 下载 bind 源代码后进行编译安装。

```
cd/tmp
tar xvfz bind-9*.tar.gz
cd bind-9*
./configure
make
make install
```

2）准备 chroot 环境

首先要以 root 身份创建下面的目录结构。

```
/chnamed
+ --dev
+ --etc
|+ --named
```

```
+ --var
+ --run
```

然后需要在 chroot 目录里面建立标准的/dev/null 设备。

```
# mknod/chnamed/dev/null c13
#cp -a/etc/localtime/chnamed/etc
```

最后要修改/etc/rc.d/init.d/syslog 中的 start 部分，改 daemon 为下面的行：

```
daemon syslogd$SYSLOGD_OPTIONS -a\/chnamed/dev/log
/etc/rc.d/init.d/syslog restart
```

2. 配置

1）配置 rndc

rndc 是 bind 的管理程序，用来启动/停止和重新装入配置文件。下面的命令把密钥串放入生成的 rndc.conf 和 named.conf 文件中。

```
# /usr/local/sbin/dnssec-keygen -ahmac-md5 -b128 -n user rndc

 [root@ cs etc]#cat rndc.conf
options {
      default-server localhost;
      default-key"rndckey";
};
server localhost {
       key"rndckey";
};
key"rndckey" {
       algorithm   hmac-md5;
       secret
"ASzhbwYZHSdiZM0CKsZONEEPzySwSivMeecSwikNVKLJBsDHBTgBcqNıWmjC";
};

 [root@ cs etc]# cat rndc.key
key"rndckey" {
       algorithm   hmac-md5;
       secret
"ASzhbwYZHSdiZM0CKsZONEEPzySwSivMeecSwikNVKLJBsDHBTgBcqNiWmjC";
};
```

2）配置 named.conf

下面是一个样例，其中假设内部网络使用 192.168.0.0/24。

```
aclournets {127.0.0.1;192.168.0.0/24};
options {
directory"/etc/named";
```

```
pid-file"/var/run/named.pid";
statistics-file"/var/run/named.stats";
allow-recursion {ournets;} ;
     } ;
controls {
inet 127.0.0.1 allow {localhost;} keys {rndc key;} ;
} ;
      //
zone"."IN {
      type hint;
      file"named.ca";
      } ;

zone"localhost"IN {
      type master;
file"localhost.zone";
      allow-update {none;} ;
      } ;
zone"0.0.127.in-addr.arpa"IN {
      type master;
      file"named.local";
      allow-update {none;} ;
} ;

zone"0.168.192.in-addr.arpa"IN {
      type master;
      file"192.168.0";
      } ;

zone"ly.com" {
      type master;
      file"named.ly";
} ;
include"/etc/rndc.key";
```

3）准备启动 named

将 named.conf 复制成/chnamed/etc/named.conf，其他数据文件复制到/chnamed/etc/named/，运行命令/usr/sbin/ntsysv，禁止自动启动 named，将如下命令加入/etc/rc.d/rc.local。

```
/usr/local/sbin/named -u named -t/chnamed -c/etc/named.conf
```

4）目录属性与用户权限设置

```
#chown named:named/chnamed/var/run
#chown named:named/chnamed
#chmod 700/chnamed
```

3. named 启动与管理

```
#/usr/local/sbin/named -u named -t/chnamed -c/etc/named.conf
#启动named
```

```
#/usr/local/sbin/rndc -slocalhostreload       #重装配置文件
#/usr/local/sbin/rndc -slocalhoststop         #停止named后台
#/usr/local/sbin/rndc -slocalhoststats        #将localhost的统计信息写入文件
#ps ax和/usr/sbin/tcpdumpport53 -n            #检查named是否处于服务状态
```

5.4.4　代理服务器 socks

socks 代理也是一种处理安全性和灵活性之间平衡的重要手段。许多情况下，需要在防火墙上钻一个洞出来而不危及整个安全策略，这时候就需要代理服务器的支持，而 socks 就是一种比较常用的代理技术。目前 socks 有两个版本，这里简单介绍一下 socks5。

1．socks5 的安装

1）软件获取

下载地址：www.socks.nec.com/cgi-bin/download.pl

2）编译安装

编译安装以目前最新版 socks5 v1.0 release 11-UNIX Source 为例。

```
#tar xvzf socks5-v1.0r11.tar.gz
#cdsocks5-v1.0r11
#./configure
#make
#make install
```

2．socks5 的配置

主要配置文件是 socks5.conf，该文件路径可在编译 socks5 时自行指定，默认路径为/etc/socks5.conf。该配置文件的内容构成如表 5-23 所示。

表 5-23　socks5.conf 文件的内容构成表

名　称	语　法	说　明
access control	permit auth cmd src-host dest-host src-port dest-port ［user-list］deny auth cmd src-host dest-host src-port dest-port ［user-list］	进行客户访问控制
authentication	auth source-host source-port auth-methods	对来自 source-host：source-port 的客户连接使用 auth-methods 定义的用户认证方法；对未定义认证方法的客户使用任何可用的认证方法
ban host	ban source-host source-port	拒绝来自 source-host：source-port 的客户连接
interface	interface hostpattern portpattern interface- address	由 interface-address 处理来自 source-host：source-port 的客户连接
proxies	proxy-type dest-host dest-port proxy-list	当客户目的请求为 dest-host：dest-port 时，使用 proxy-list 中的代理服务器请求数据
variables	set variable value	定义 socks5 运行参数

3. 配置示例

假定架设一个有 50 台计算机和一个有 32 个（5 位）IP 地址，有多重访问等级的子网。这些等级按照从低到高分别为 b2、b1、b0。

1）网络设计

192.168.2.255 用作广播；

32 个 IP 位址挪出 23 个，提供给进行 Internet 访问的机器；

一个 IP 给 Linux box；

另一个 IP 给另一个 Linbux box；

两个 IP 号码给 Router；

4 个 Domain Names 设为 paul、ringo、john 和 george，用来掩人耳目；

保留地址为 192.168.2.xxx；

建立两个保护的分离的子网，两个子网各接一台 Linux box；

一个文件服务器用不同的网卡连接两个保护网路，关闭 IP forwarding，对 b1 级用 192.168.2.17，对 b0 级用 192.168.2.23。

2）Proxy 的架设

b2 网直接连 Internet，b1 网及 b0 网已在防火墙内，所以 b2 网中不用架设 Proxy Server，b1 网和 b0 网的架设十分相似。下面是应用的规则。

- 不许任何人用文件服务器进行因特网访问。
- 不允许 b1 使用浏览 Web，但仍开放其他服务。
- 除已被拒者之外，192.168.2.xxx 均可以使用该 Proxy Server（例如文件服务器和 b1 网的 Web 访问）。
- b1 网 Linux box 上的 sockd.conf 设定如下。

```
deny 192.168.2.17 255.255.255.255
deny 0.0.0.0 0.0.0.0 eq 80
permit 192.168.2.0 255.255.255.0
```

- b0 网 Linux box 上的 sockd.conf 设定如下。

```
deny 192.168.2.23 255.255.255.255
permit 192.168.2.0 255.255.255.0
```

5.4.5 邮件服务器

Linux 下常用的邮件服务器主要是 sendmail、Postfix 和 qmail。这里选择 qmail。qmail 是一种可以完全替代 Sendmail-binmail 体系的新一代 UNIX 邮件系统，较之其他邮件服务器软件具有安全、可靠、高效和简单的特点，由于支持 Maildir，保证了系统在突然崩溃的情况下不至于破坏整个信箱。

1. relay 管理

qmail 中有一个决定是否接收某个邮件的配置文件 rcpthosts。只有当接收者地址的域名

出现在 rcpthosts 文件时，才接收该邮件，否则拒绝。若该文件不存在，默认接收所有的邮件。显然，没有 rcpthosts 的 qmail 服务器是开放转发的。

设置自己的服务器为非开放转发的最简单办法，就是将邮件服务器的所有域名（若 DNS 的 MX 记录指向该机器，也应该包括该机器的其他域名）放入 rcpthosts。不过，这样就拒绝了除了本地机器之外的任何客户机使用你的服务器转发邮件，而要支持客户使用 MUA 来发送邮件，必须允许客户使用服务器转发邮件。qmail-smtpd 支持一种有选择性的忽略 rcpthosts 文件的方法：若 qmail-smtpd 的环境变量 RELAYCLIENT 被设置，则忽略 rcpthosts 文件，允许 relay。qmail 通过判断发送邮件者的源 IP 地址，来确定该发送者是否为自己的客户。这里用到的是 tcpserver 程序。

tcpserver 的配置文件是/etc/tcp.smtp，该文件定义了是否对某个网络设置 RELAYCLIENT 环境变量。例如，本地网络地址为 192.168.0.0/24，若连接来自 127.0.0.1 和 192.168.0 则允许，且为其设置环境变量 RELAYCLIENT；否则允许其他连接但不设置 RELAYCLIENT 环境变量。这样从其他地方到本地的 25 号连接将被允许，但由于未设环境变量，其连接被 qmail-smptd 所拒绝。tcp.smtp 的内容应设置如下：

```
127.0.0.1:allow, RELAYCLIENT="
192.168.0.:allow, RELAYCLIENT="
:allow
```

但是 tcpserver 并不直接使用/etc/tcp.smtp 文件，而是先要将该文件转化为 cbd 文件。

```
[wly@ cs/etc] $#tcprules tcp.smtp.cdbtcp.smtp.temp<tcp.smtp
```

在/service/qmail-smtpd 目录下的 run 文件中有/usr/local/bin/tcpserver -v -p -x/etc/tcp.smtp.cdb。可以看到，tcpserver 利用了/etc/smtp.cbd 文件。若本地有多个网络，则需要这些网络都出现在/etc/tcp.smtp 文件中。这样就实现了允许本地客户 relay 邮件，而防止 relay 被滥用。

2. 带 smtp 身份认证的 qmail

smtp 身份认证就是让用户在发送邮件之前首先提供用户名和密码。为了使 qmail 支持身份认证，需要几个附加软件包，连同标准的 qmail 程序包，需要下载以下几个程序包：

```
checkpassword-0.90.tar.gz或qmail-smtpd.c
cmd5checkpw-0.22.tar.gz
ucspi-tcp-0.88.tar.gz
qmail-1.03.tar.gz
vpopmail-4.9.10.tar.gz
```

1）下载后进行安装

```
tar zxvfcheckpassword-0.90.tar.gz
cd checkpassword-0.90
make
make setup check
```

```
tar zxvf cmd5checkpw-0.22.tar.gz
mkdir/usr/man
mkdir/usr/man/man8
cd cmd5checkpw-0.22
make
make install

tar zxvf ucspi-tcp-0.88.tar.gz
cd ucspi-tcp-0.88
make
make setup check

cp qmail-smtpd.c qmail-1.03/
```

2）添加用户

```
mkdir/var/qmail
groupaddnofiles
useradd -g nofiles -d /var/qmail/alias alias
useradd -g nofiles -d /var/qmail qmaild
useradd -g nofiles -d /var/qmail qmaill
useradd -g nofiles -d /var/qmail qmailp
groupadd qmail
useradd -g qmail -d /var/ qmail qmailq
useradd -g qmail -d /var/ qmail qmailr
useradd -g qmail -d /var/ qmail qmails
```

3）进行 qmail 的基本配置，首先要用自动配置程序进行基本的操作

```
cd qmail-1.03
make setup check
./config-fast cs.ly.com
cd~alias
touch.qmail-postmaster.qmail-mailer-daemon.qmail-root
chmod644~alias/.qmail*
echo"127.0.0.1:allow, RELAYCLIENT=""">/etc/tcp.smtp

#checkpassword程序需要访问/etc/shadow，所以需要setuid
chmod 4755/bin/checkpassword

cp/var/qmail/boot/home/var/qmail/rc
```

修改/var/qmail/rc，把./Mailbox 改成./Maildir/，使用 Maildir。

4）启动 qmail

```
csh -cf"/var/qmail/rc &"
```

5）启动 smtp 服务

```
tcpserver -H -R -l 0 -t 1 -c 100 -x /etc/tcp.smtp.cdb -u 507 -g 502 0 smtp
```

```
/var/qmail/bin/qmail-smtpd/bin/check   password/bin/true/bin/md5checkpw/
bin/true &
```

注意：使用 checkpassword 验证，-u 507 -g 502 指的是 qmail 和 nofiles 的。

这样基于系统用户的 smtp 验证就完成了。

3．基于 vpopmail 的实现

若在 qmail 系统中使用 vpopmail，可利用 vpopmail 针对漫游用户的配置选项来防止邮件系统的 relay 功能被滥用。其支持漫游用户的原理是：当漫游用户通过 POP3 取信后的某段时间内允许该地址通过邮件服务器转发信件，而经过一段时间不活动之后，这个 relay 功能就被禁止了。

1）使用 vpopmail

```
//建立vpopmail的用户和组
groupaddvchkpw
useradd -g vchkpw -d /vmail vpopmail
//切换到vpopmail用户
su vpopmail
//建立用户子目录
mkdir ~vpopmail/etc
//建立访问控制文件
echo"127.0.01.:allow, RELAYCLIENT="""">~vpopmail/etc/tcp.smtp
//编译安装
./configure --enable-default-domain=cs.ly.com --enable-roaming-users=y
make
make install-strip
//改变当前目录
cd ~vpopmail/bin
//添加域用户
./vadddomaincs.ly.com
./vadduseryyy@ cs.ly.com
//更改文件属性
chmod 6755 /vmail/bin/vchkpw
```

2）启动 smtp 服务

```
/var/qmail/bin/qmail-smtpd/vmail/bin/vchkpw/bin/true/bin/md5checkpw/bin
/true &
```

3）启动 POP3 服务

```
tcpserver -H -R 0pop-3/var/qmail/bin/qmail-popup cs.ly.com/vmail/bin/vchkpw
/var/qmail/bin/qmail-pop3dMaildir &
```

&表示后台执行。

注意：使用 mysql 等其他模块验证时不影响。

第 6 章

路由器安全管理

在目前的网络体系中，路由器是多种网络互联的重要设备，因为路由器一般位于防火墙之外，是边界网络的前沿，所以路由器的安全管理成为了第一道防线。在默认情况下，路由器访问密码存储在固定位置，用第一章讲到的 sniffer 嗅探器很容易获得登录名和密码，从而使路由器完全受到攻击者控制，从而入侵整个路由器管理的网络。目前的路由器种类繁多，优质的路由器都有自己丰富的安全机制，一般都内置了入侵检测系统，但还需要网络管理员配置相应的安全策略及进行相应的管理。在这一章中，针对路由器使用最多的 AAA（验证、授权和审计）、访问控制技术、数据加密和防伪和数据加密技术（VPN 技术）进行系统介绍，使管理员有章可循，路由器平台很多，国内应用最多的主要有思科公司的 IOS 平台和华为公司的 VRP 平台两大阵营，本章以华为的 VRP 的安全配置命令为例进行介绍。

6.1 路由器安全概述

路由器相关安全特性具有两层含义：保证内部局域网的安全（不被非法侵入）和保护外部进行数据交换的安全。路由器安全关注的范围包括保护网络物理线路不会轻易遭受攻击、有效识别合法的用户和非法的用户、实现有效的访问控制、保证内部网络的隐蔽性、有效的防伪手段、重要的数据重点保护、对网络设备、网络拓扑的安全管理，对病毒提高安全防范意识。在开放式的网络环境中，每个网络都是一种对等关系，相互之间可以直接访问。为了增强网络的安全性，需要将这种对等界定在一定的范围之内。将一个网络划分为多个部分，在同一个网络中的某些主机可以认为是“互相信任的”，而和其他的主机处于一种“不信任关系”，使开放的环境处于一种受控的状态。同时信息加密也是保证数据安全的一种十分重要的手段。产生网络安全事故的很多原因是人为因素，例如安全意识淡漠、有意利用自己的某些特权等。因此保护网络的安全除了要进行技术上的更新之外，同时还需要对员工进行必要的安全意识教育。

针对网络存在的各种安全隐患，路由器必须具有的安全特性包括身份认证、访问控制、信息隐藏、数据加密和防伪、安全管理、可靠性和线路安全。可靠性要求主要针对故障恢复、负载能力和主设备运行故障时，备份自动接替工作。负载分担主要指网络流量增大时，

备份链路承担部分主用链路的工作，线路安全指的是线路本身的安全性，用于防止非法用户利用线路进行访问。网络安全身份认证包括访问路由器时的身份认证、Console 登录配置、Telnet 登录配置 、SNMP 登录配置、Modem 远程配置、对其他路由的身份认证、直接相连的邻居路由器配置、逻辑连接的对等体配置、路由信息的身份认证、防伪造路由信息的侵入安全特性。

身份认证是网络安全中解决的一个重要的问题，主要保证只有合法的用户和经过授权的用户才可以访问、控制路由器。如需要配置路由器时，需要验证用户名和密码。同时还需要保证和其他网络设备的信息交互具有合法的身份认证，如防伪造路由信息的侵入等。因此在一些需要路由器重要信息的场合，都需要进行身份验证，来保证信息来源的可靠。路由器安全技术包括 AAA（Authentication、Authorization 和 Accounting），它是验证、授权和记账的简称。网络安全服务提供一个实现身份认证的框架来提供验证、授权、记账服务，使用 RADIUS 等协议实现对网络的访问控制。AAA 技术可以提供基于用户的验证、授权、记账服务。基于用户的含义是，AAA 技术不是根据 IP 地址等信息来验证用户，而是根据用户名、口令对用户进行验证。AAA 技术主要使用在拨号接入访问上，用户利用电话拨号入网就是依靠 AAA 技术来实现验证、授权和计费的。因此在接入服务中，AAA 是一个最有效实用的安全手段。RADIUS 采用客户机/服务器（Client/Server）结构。验证、授权时客户端的任务是将用户（User）的信息发送到指定的服务器，然后根据服务器的不同的响应进行相应处理。RADIUS 服务器的任务是接收客户端发来的用户连接请求，然后验证用户并返回客户端提供服务所需要的配置信息。RADIUS 服务器的数据库中集中存放了相关的安全信息，避免安全信息凌乱散布带来的不安全性，同时更可靠且易于管理。实现计费时，客户端将用户的上网时长、进出字节数、进出包数等原始数据送到 RADIUS 服务器上，以供 RADIUS 服务器计费时使用。

访问控制实现如下功能：对网络设备的访问控制，分级保护不同级别的用户拥有不同的操作权限，基于五元组（指 IP 包头中的源 IP 地址、目的 IP 地址、协议号、源端口和目的端）的访问控制，根据数据包信息进行数据分类，不同的数据流采用不同的策略，基于用户的访问控制，对于接入服务用户，设定特定的过滤，访问控制是路由器提供的一种重要的安全策略，访问控制可以有效地防止一些非法的访问。包过滤技术是指提供访问控制的基本框架，来提供基于 IP 地址等信息的包过滤、提供基于接口的包过滤和提供基于时间段的包过滤，包过滤技术是利用访问控制列表实现的一种防火墙技术。包过滤技术是最常用的访问控制手段，包过滤技术最显著的特点是利用 IP 数据包的特征进行访问控制，不像 AAA 技术那样是根据用户名、密码进行访问控制。因此包过滤技术不能使用于接入服务中，它适用于用户根据 IP 地址、端口等定义合适的规则，阻止对网络直接的非法访问。利用包过滤技术可以阻挡“不信任网络”的访问。例如，某个内部局域网接入了 Internet，这个局域网只信任它的一个分公司的网络和某些重要客户的网络。在 Internet 上，每个公司的 IP 地址是不会经常改变的，因此利用包过滤技术，就可以限制除了分公司、重要客户以外的其他的网络对内部局域网的访问。

信息隐藏实现功能包括地址转换以隐藏内网的内部地址、内部用户可以直接发起建立连接请求来保护内部局域网访问 Internet。地址转换技术主要使用在内部局域网对公有网络的访问。使用地址转换技术不仅可以使用许多局域网用户可以共享一个 IP 地址上网，而且

可以使内部局域网的网络结构、IP 地址等信息都不在 Internet 上暴露，增强了内部局域网的安全特性。地址转换技术主要使用在一个局域网公用上网的情况。地址转换技术同时隐藏了内部局域网的真实 IP 地址和网络拓扑，转换内部局域网为外部的一个 IP 地址或少量 IP 地址（地址池），使 Internet 上的其他网络不知内部局域网的真实的 IP 地址和网络拓扑，保护了内部局域网的安全，同时又不影响路由器。地址转换能够将网内用户发出报文的源地址全部映射成一个接口地址。与按需拨号相结合，使局域网内用户通过一台路由器即可轻松上网。

数据加密和防伪技术是利用公网传输数据不可避免地面临数据窃听的问题，传输之前进行数据加密，保证只有与之通信的接收端才能够解密数据，防伪报文在传输过程中，被截获、修改，重新投放到网络时，接收端可以进行数据识别、丢弃被修改的报文。相关技术包括：数据加密技术、数字签名技术、IPSec 协议及相关技术。数据加密技术主要是将需要在 Internet 上传递的数据加密。加密技术包含两个方面：普通的加密和防伪。防伪技术可以防止报文被不法分子截获之后，将报文修改，然后重新放到网上继续传递。数据加密防伪是保护 Internet 上数据安全的一个重要手段，利用这种技术可以在 Internet 上为用户提供一种“安全的 VPN”服务，利用加密技术可以为用户提供一种安全的在 Internet 上传递数据的手段。路由器提供的 IPSec 和 IKE 技术，IPSec（IP Security）可以实现数据的加密以及防伪，可以使在不安全的线路上传输加密信息形成“安全的隧道”。可以为用户在 Internet 提供安全的 VPN 解决方案。IKE（密钥交换协议）为通信双方提供交换密钥等服务，IKE 定义了通信双方进行身份认证、协商加密算法以及生成共享的会话密钥的方法。并且保证永远不在不安全的网络上直接传送密钥，而是通过一系列交换信息计算密钥。IPSec 和 IKE 技术的结合使用，有效地提供了在 Internet 网络上进行数据加密、数据防伪的功能。IPSec（IP Security）是一组开放协议的总称，特定的通信方之间在 IP 层通过加密与数据源验证，以保证数据包在 Internet 网上传输时的私有性、完整性和真实性。IPSec 通过 AH （Authentication Header）和 ESP（Encapsulating Security Payload）这两个安全协议来实现。此实现不会对用户、主机或其他 Internet 组件造成影响，用户还可以选择其他的硬件和软件加密算法，而不会影响其他部分的实现。

Internet 密钥交换协议（IKE）用于通信双方协商和建立安全联盟，并交换密钥。IKE 定义了通信双方进行身份认证、协商加密算法以及生成共享会话密钥的方法。IKE 的精髓在于它永远不在不安全的网络上直接传送密钥，而是通过一系列数据的交换，通信双方最终计算出共享的密钥，并且即使第三方截获了双方用于计算密钥的所有交换数据，也不足以计算出真正的密钥。

虚拟私有网（Virtual Private Network，VPN）是近年来随着 Internet 的发展而迅速发展起来的一种技术。现代企业越来越多地利用 Internet 资源来进行促销、销售、售后服务、培训和合作等活动。许多企业趋向于利用 Internet 来替代他们的私有数据网络。相对于企业原有的 Intranet，这种利用 Internet 的虚拟链路来传输私有信息而形成的逻辑网络就称为虚拟私有网。VPN 的一个核心技术就是“隧道技术”，这种技术的主要思想是将一种类型网络的数据包通过另一种类型网络进行传输。二层隧道是建立在链路层的隧道，三层隧道是建立在网络层的隧道。

安全管理是指保证重要的网络设备处于安全的运行环境，防止人为破坏、保护访问口

令、密码等重要的安全信息、进行安全策略管理，有效利用安全策略，在网络出入口实现报文审计和过滤，提供网络运行的必要信息。对路由器等重要网络设备的管理是保证路由器安全运行的一个重要方面，一定要保证没有权限的用户不能随便配置路由器，也不能得到路由器的配置信息。网络安全同样需要保障网络拓扑信息的安全。安全接入 Internet 包括基于接口的包过滤、基于时间段定义过滤规则、通过地址转换访问 Internet、外部不能直接访问内部网络，可以通过地址转换向外提供 WWW、FTP 等服务，避免内部服务器直接受到攻击，日志主机可以记录网络运行情况便于用户的安全分析与管理。

通用路由平台（Versatile Router Platform，VRP）是华为系列路由平台的简称，是整个 VRP 平台的核心，它实现了 OSPF、BGP、IS-IS、RIP、EIGRP、PIM DM/SM 等多种单播和多播路由协议，支持路由迭代、路由策略、路由聚合等丰富的路由特性，提供了完整的路由功能。但路由器必须防范来自公网上的恶意攻击。VRP 的安全特性如下：

- 基于 RADIUS（Remote Authentication Dial-In User Service）的 AAA 服务与 RADIUS 服务器配合实施的 AAA 服务，可以提供对接入用户的验证、授权和计费安全服务，防止非法访问。
- 验证协议：在 PPP 线路上支持 CHAP 和 PAP 验证。
- 包过滤（Packet Filter）：用访问控制列表实现，允许指定可以通过（或禁止通过）路由器的报文类型。
- 应用层报文过滤 ASPF（Application Specific Packet Filter）：也称为状态防火墙，是一种高级通信过滤，它检查应用层协议信息并且监控基于连接的应用层协议状态，维护每一个连接的状态信息，并动态地决定数据包是否被允许通过防火墙或者被丢弃。
- 网络层安全（IP Security，IPSec）：特定的通信方之间在 IP 层通过加密与数据源验证，来保证数据包在 Internet 上传输时的私有性、完整性和真实性。
- 事件日志：记录系统安全方面事件，实时跟踪非法侵入。
- 地址转换：NAT 网关将公共网络和企业内部网分隔离开来，在公共网络中隐藏企业内部设备的 IP 地址，阻止来自公共网络上的攻击。
- 相邻路由器验证：确保所交换路由信息的可靠性。
- 视图分级保护：将用户分成 4 级，每级用户赋予不同的配置权限，级别低的用户不能进入更高级的视图。系统命令行采用分级保护方式，命令行划分为参观级、监控级、配置级和管理级 4 个级别，只有提供了正确的登录口令，才能使用相应的命令。

6.2 AAA 与 RADIUS 协议原理及配置

在这一小节中介绍 AAA 与 RADIUS 协议的原理、配置方法及调试及典型配置举例。首先介绍 AAA 与 RADIUS 协议原理。

6.2.1 AAA 与 RADIUS 协议原理

AAA 是 Authentication（认证）、Authorization（授权）和 Accounting（计费）的简称。

它提供对用户进行认证、授权和计费三种安全功能。AAA 一般采用客户/服务器结构，客户端运行于被管理的资源侧，服务器上则集中存放用户信息。这种结构既具有良好的可扩展性，又便于用户信息的集中管理。具体如下：

- 认证：认证用户是否可以获得访问权，确定哪些用户可以访问网络。
- 授权：授权用户可以使用哪些服务。
- 计费：记录用户使用网络资源的情况。

实现 AAA 功能可以在本地进行，也可以由 AAA 服务器在远程进行。计费功能由于占用系统资源大通常都使用 AAA 服务器实现。对于用户数量大的情况，验证和授权也应该使用 AAA 服务器。AAA 服务器与网络设备的通信有标准的协议，目前比较流行的是 RADIUS 协议。

提供 AAA 支持的服务包括：PPP 的 PAP 和 CHAP（验证用）、通过 telnet 登录到路由器，以及通过各种方式（如 console 口，aux 口等）进入到路由器进行配置的操作和 FTP 即通过 ftp 登录到路由器的用户，如图 6-1 所示。

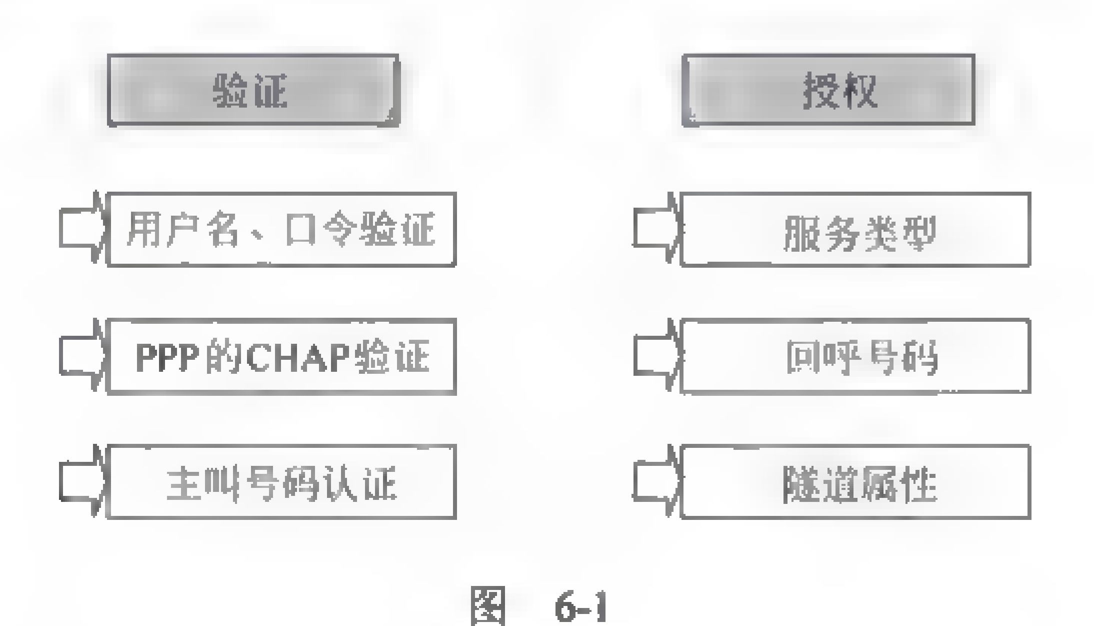


图 6-1

1. 验证

用户名、口令验证：包括 PPP 的 PAP 验证、用户的 CHAP 验证、EXEC 用户验证、FTP 拥护验证和拨号的 PPP 用户可以进行号码验证。

2. 授权

服务类型授权包括一个用户授权提供的服务。可以是 PPP、EXEC 和 FTP 中的一种或几种。回呼号码对 PPP 回呼用户可以设定回呼号码。隧道属性配置 L2TP 的隧道属性。验证、授权可以在本地进行，也可以在 RADIUS 服务器进行。但对一个应用服务的验证和授权应使用相同的方法，可以使验证、授权均在本地进行，也可以使用 RADIUS 服务器。

RADIUS 是远程认证拨号用户服务（Remote Authentication Dial-In User Service）的简称，最初由 Livingston Enterprise 公司开发，作为一种分布式的客户机/服务器系统，能提供 AAA 功能。RADIUS 技术可以保护网络不受未授权访问的干扰，常被用在既要求较高安全性又要求维持远程用户访问的各种网络环境中（如用来管理使用串口和调制解调器的大量分散拨号用户）。

RADIUS 基于客户/服务器模型，NAS（如路由器）作为 RADIUS 客户端，负责传输用户信息到指定的 RADIUS 服务器，然后根据从服务器返回的信息进行相应处理（如接入 /

挂断用户）。RADIUS 服务器负责接收用户连接请求，认证用户，然后给 NAS 返回所有需要的信息。RADIUS 服务器通常要维护三个数据库：第一个数据库 Users 用于存储用户信息(如用户名、口令及使用的协议、IP 地址等配置)，第二个数据库 Clients 用于存储 RADIUS 客户端的信息(如共享密钥)，第三个数据库 Dictionary 存储的信息用于解释 RADIUS 协议中的属性和属性值的含义。如图 6-2 所示。

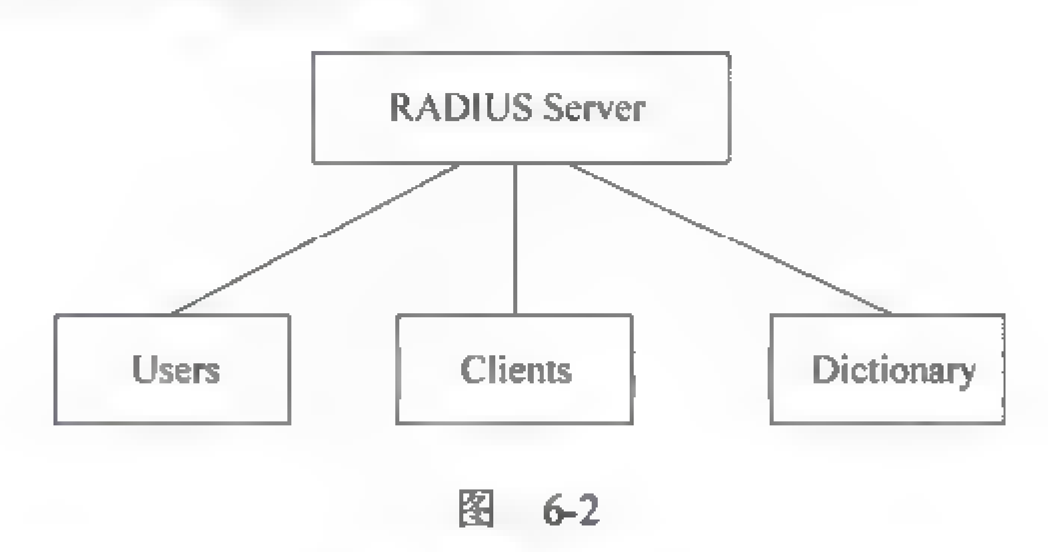


图　6-2

另外，RADIUS 服务器还能够作为其他 AAA 服务器的客户端进行代理认证或计费。RADIUS 服务器支持多种方法来认证用户，如基于 PPP 的 PAP、CHAP 认证，基于 UNIX 的 Login 等。

RADIUS 服务器对用户的认证过程通常需要利用 NAS 等设备的代理认证功能，RADIUS 客户端和 RADIUS 服务器之间通过共享密钥认证相互间交互的消息，用户密码采用密文方式在网络上传输，增强了安全性。RADIUS 协议合并了认证和授权过程，即响应报文中携带了授权信息。RADIUS 的基本消息交互流程如图 6-3 所示。

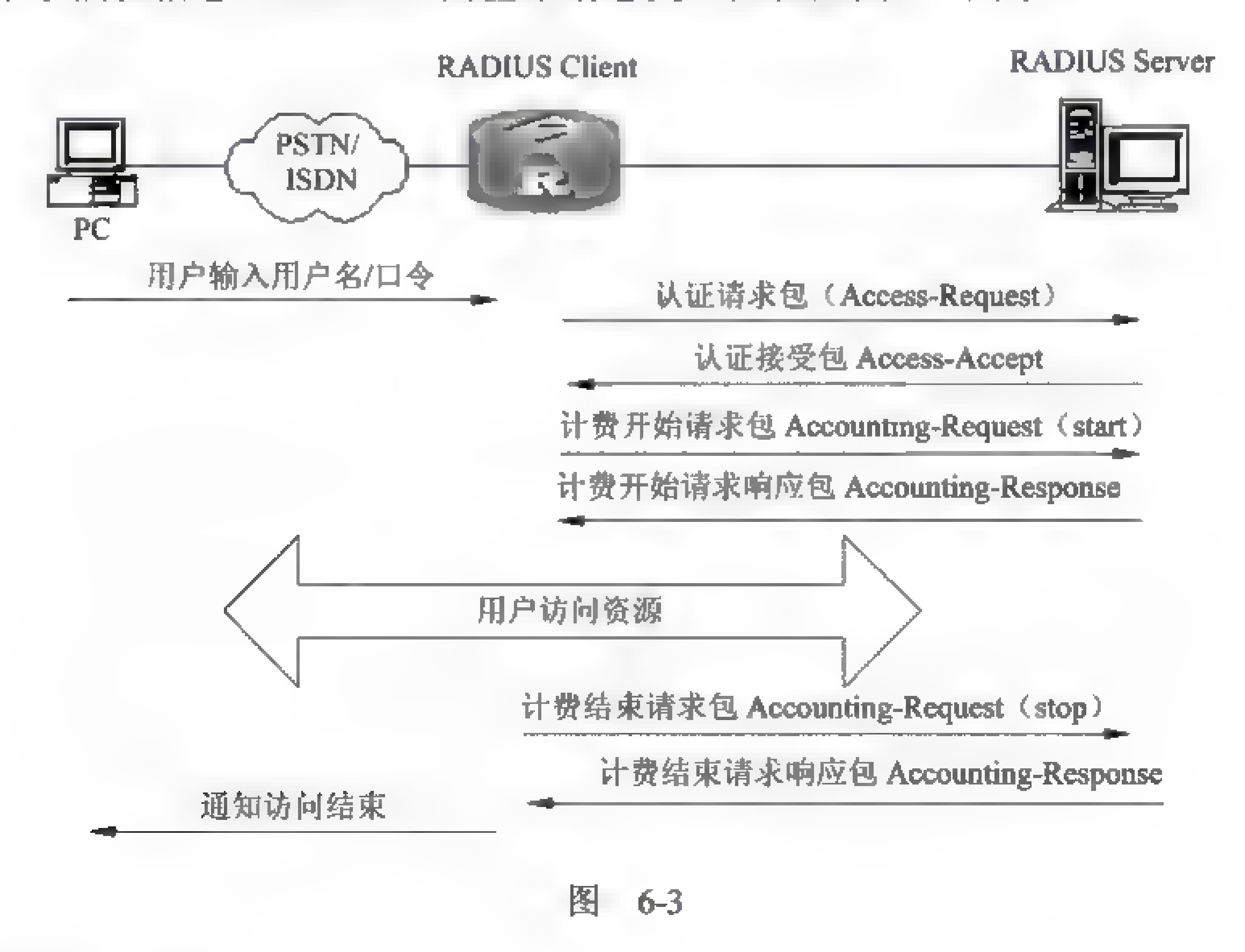


图　6-3

基本交互步骤如下：

（1）用户输入用户名和口令。

（2）RADIUS 客户端根据获取的用户名和口令，向 RADIUS 服务器发送认证请求包(Access-Request)。

（3）RADIUS 服务器将该用户信息与 Users 数据库信息进行对比分析，如果认证成功，则将用户的权限信息以认证响应包（Access-Accept）发送给 RADIUS 客户端；如果认证失败，则返回 Access-Reject 响应包。

（4）RADIUS 客户端根据接收到的认证结果接入/拒绝用户。如果可以接入用户，则 RADIUS 客户端向 RADIUS 服务器发送计费开始请求包（Accounting-Request），Status-Type 取值为 start。

（5）RADIUS 服务器返回计费开始响应包（Accounting-Response）。

（6）RADIUS 客户端向 RADIUS 服务器发送计费停止请求包，Status-Type 取值为 stop。

（7）RADIUS 服务器返回计费结束响应包（Accounting-Response）。

RADIUS 采用 UDP 传输消息，通过定时器管理机制、重传机制、备用服务器机制，确保 RADIUS 服务器和客户端之间交互消息的正确收发。RADIUS 报文结构如图 6-4 所示。

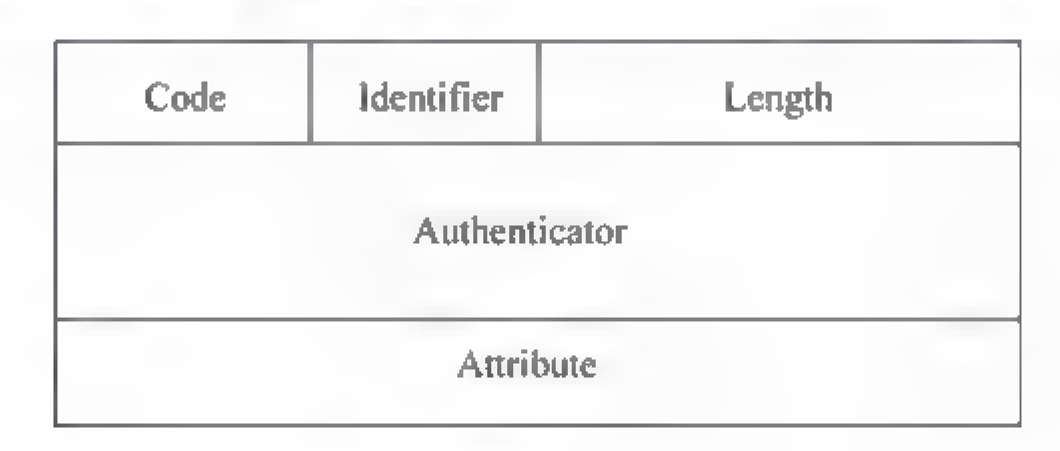


图 6-4

其中 Identifier 域用于匹配请求包和响应包，随着 Attribute 域改变、接收到有效响应包也在不断变化，而在重传时保持不变 Authenticator 域（16 字节）用于验证 RADIUS 服务器传输回来的请求，同时用于密码隐藏算法上，分为 Request Authenticator 和 Response Authenticatoro，Request Authenticator 采用 16 字节的随机码。Response Authenticator 是对 Code、Identifier、Request Authenticator 、Length、Attribute 和共享密钥进行 MD5 算法后的结果。RADIUS 使用 UDP 作为传输协议，具有良好的实时性。同时也支持重传机制和备用服务器机制，从而有较好的可靠性。RADIUS 的实现比较简单，适用于大用户量时服务器端的多线程结构。正因为如此 RADIUS 协议得到了广泛的应用。

RADIUS 实现 AAA 的流程如图 6-5 所示。

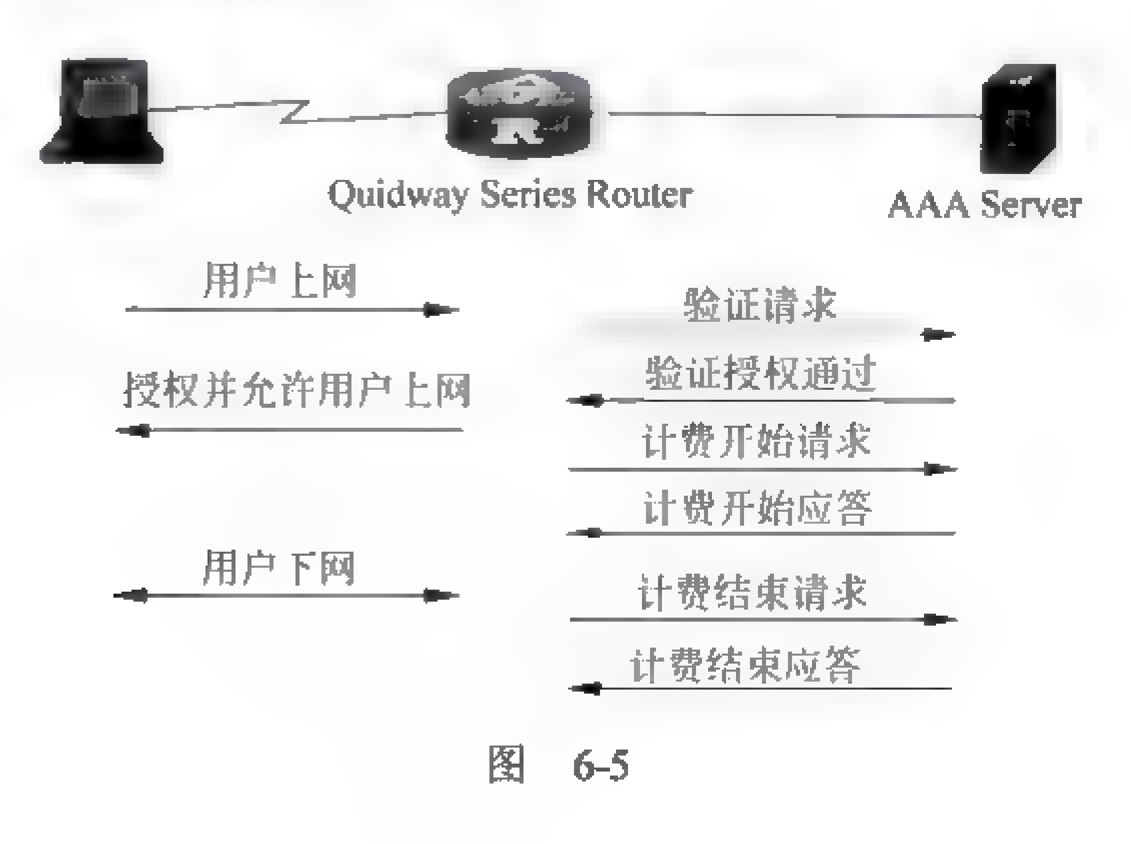


图 6-5

RADIUS 协议采用客户机/服务器结构，路由器作为客户端与 RADIUS 服务器通信的桥梁。UDP（User Datagram Protocol，用户数据报协议）是一种面向无连接的协议，传输层保证报文的可靠性和顺序性，因为报文可能丢失或者是乱序。RADIUS 协议使用了两个 UDP 端口分别用于验证（以及验证通过后对用户的授权）和计费。在 RADIUS 的协议文本 RFC 2138 和 RFC 2139 中，使用 1812 号端口作为验证端口，1813 号端口为计费端口。也可以使用其他端口。RADIUS 协议采用了“请求/响应”的操作模式，请求由客户端发起，当 RADIUS 服务器收到合法的请求后就要给予响应。由于 UDP 报文可能会丢失，网络也可能临时出现故障，因此路由器提供重传机制，当在一定时间内没有收到 RADIUS 服务器的响应时，会重传刚才的请求。如果多次重传后仍然收不到响应，那么路由器会向备用的 RADIUS 服务器发送请求。作为安全协议，RADIUS 自身的安全性也有一定考虑。客户端与服务器端有共享密钥。通信时使用 MD5 算法，通过共享密钥对包进行数字签名，验证签名的正确性可以防止网络上的其他主机冒充路由器或者 RADIUS 服务器。用户口令也需要进行加密后再在 Internet 上传送，使口令不会泄漏。RADIUS 包有 0 到多个属性，用户的各种信息均写在属性中，一些属性协议还规定了各属性值的含义。性能的扩展只需要增加包中所带的属性即可。使用中还可以定义私有的属性类型和属性值。这需要修改 RADIUS 服务器的属性字典。

RADIUS 实现 AAA 验证和授权过程如下：

（1）首先发送验证请求包。在用户名、口令验证时，验证请求包包含用户名和加密后的口令；CHAP 验证中包含用户名，CHAP 验证过程中的各项（Challenge、CHAP Identifier 和 Response）。主叫号码验证还需要有主叫号码。

（2）RADIUS 服务器收到验证请求包后，首先检查包的合法性，然后根据包中用户信息验证用户是否合法。如果用户非法，则向路由器发送访问拒绝包，如果用户合法，那么 RADIUS 服务器会将用户授权信息（如用户类型、回呼号码等）打包发送到路由器，该包称为访问接受包。

（3）路由器收到访问接受/拒绝包时，首先要判断包中的签名是否正确，如果不正确将认为收到了一个非法的包。如果签名正确，且收到的是访问接受包，那么路由器会判断授权服务类型是否与此用户相符，如果不符则拒绝该用户的上网请求，如果符合则接受用户的上网请求，并使用其他用户授权信息对用户进行处理（如回呼、L2TP 隧道属性的设置）。如果签名正确且收到的是访问拒绝包，则拒绝该用户的上网请求。

6.2.2 AAA 与 RADIUS 协议配置方法

首次使用 AAA，经常发生配置了用户而验证不通过的情况。这实际上是由于没有学会灵活使用 aaa accounting-scheme optional 的原因。这种情况不是验证不通过，而是计费失败，切断了用户。

因为开始使用的时候启用 AAA，这时默认使用本地验证。而本地验证也是需要计费的，由于没有配置 RADIUS 服务器，造成计费失败，而因为没有配置 aaa accounting-scheme optional，在计费失败时就断开用户，因此用户不能成功上网。

aaa accounting-scheme optional 的作用是在计费失败时允许用户继续使用网络。因此在

验证不了计费的情况下，一定要注意配置 aaa accounting-scheme optional 命令。

AAA 的配置包含如下几个主要步骤。

1. 首先应该能够使用 AAA

在默认情况下禁止使用 AAA。在系统视图下进行下列配置，操作命令为：

AAA aaa enable

禁止 AAA undo aaa enable

2. 配置认证方案

（1）如果配置通过 FTP、Telnet 登录到路由器，以及通过各种终端服务方式（如 Console 口、Aux 口等）进入到路由器进行配置的操作的 Login 用户认证方案，在系统视图下操作命令如下。

aaa authentication-scheme login { default | scheme-name } [method1 | [template server-template-name [method2]]]

删除 AAA 的 Login 认证方案或恢复。

undo aaa authentication-scheme login {default | scheme-name }

其中，method1 为认证方法，可以有以下 5 种情况：none、local、radius、radius none 和 radius local。method2 只能为 local 或 none。

注意：FTP、Terminal、SSH 不是 RADIUS 协议的标准属性取值，需要修改 RADIUS 服务器的属性，在属性 login-service（标准属性 15）中增加了两个取值的定义：

login-service(50) = FTP

login-service(51)= Terminal

login-service(52)= SSH

修改后再启动 RADIUS 服务器方可。

（2）如果配置通过与路由器或接入服务器建立 PPP 连接（例如拨号、PPPoE 和 PPPoA 等）从而访问网络的用户的 PPP 用户认证方案，在系统视图下操作命令如下。

aaa authentication-scheme ppp { default |scheme-name } [method1 | [template server-template-name [method2]]]

禁止用指定方案。

undo aaa authentication-scheme ppp { default | scheme-name }

其中 method1 和 method2 与配置 Login 的参数相同，另外可以配置多个 PPP 的认证方案，用于不同端口。

（3）如果配置 AAA 的本地优先认证，也就是在未配置本地优先认证时，先对用户进行 RADIUS 认证。在系统视图下操作命令如下。

aaa authentication-scheme local-first

不使用本地优先认证。

undo aaa authentication-scheme local-first

默认为不使用本地优先认证。

3．配置计费方案

1）配置 AAA 的计费可选

打开或关闭 AAA 计费可选开关。在对用户计费时如果发现没有可用的 RADIUS 计费服务器或与 RADIUS 计费服务器通信失败时，若配置了 aaa accounting-scheme optional 命令，则用户可以继续使用网络资源。否则用户的连接将被切断。计费均是通过独立计费服务器实现，本地不提供计费功能。

请在系统视图、radius template 视图进行下列配置：

打开计费可选开关 aaa accounting-scheme optional

关闭计费可选开关 undo aaa accounting-scheme optional

计费可选开关配置对所有非指定 RADIUS 服务器或服务器模板的计费均起作用。

2）配置 AAA 的 PPP 计费方案

对通过与路由器或接入服务器建立 PPP 连接，从而访问网络的用户，进行计费时使用 PPP 计费方案。在系统视图下进行下列配置：

PPP 计费方案配置操作命令对使用 PPP 服务的用户按指定方案进行计费。

aaa accounting-scheme ppp { default | scheme-name } { [start-stop | wait-start | stop-only] { template server-template-name | radius } | none }

取消计费方案或恢复默认计费方案的计费方法。

undo aaa accounting-scheme ppp { default | scheme-name }

4．配置本地用户数据库

本地数据库用于在本地记录用户的相关属性列表，如用户名、口令、等级及其他相关信息，它可以直接用于 FTP、Telnet、Terminal 和 PPP 用户的本地认证（即不启动 AAA，具体配置请分别参见相关章节），也可以用作 AAA local 认证方法的用户数据库。

当用户配置了 radius 认证方法时，应在 radius 服务器端进行类似的配置（是否能配置及如何配置取决于采用的服务器），此时本地配置将不起作用。

1）配置用户名及口令

此命令用于配置本地用户数据库中的用户名及口令。用户数据库不仅用于 login 用户（包括 Telnet、FTP 及 Terminal 用户等）认证，还可以用于 PPP 用户的认证。在系统视图下进行配置操作命令如下：

配置用户及口令 local-user local-user [password { simple | cipher } password]

取消用户 undo local-user local-user

simple 代表明文，当使用 display 命令显示用户信息时，该口令以明文显示。

cipher 代表密文，当使用 display 命令显示用户信息时，该口令以密文显示。

如果仅输入 local-user local-user 命令后即回车，系统不会做任何反应，即不会生成用户名为 local-user 的新用户，对原有用户名为 local-user 用户也不会有任何影响。

2）授权用户可以使用的服务

本命令用于对本地用户可以使用的服务进行授权。默认为授权所有服务。在系统视图下进行下列配置：

配置用户授权 local-user local-user service-type { [pad] [ftp] [ppp] [terminal] [telnet] [ssh] }

恢复用户默认授权 undo local-user local-user service-type

3）设置用户的优先级

系统提供对命令的分级管理，即对所有命令设定一定的操作级别，只有用户的优先级等于或高于命令的级别，用户才有使用该命令的权限。在系统视图下进行下列配置：

配置用户的优先级 local-user local-user level level

设置用户的优先级为默认值 undo local-user local-user level

level 代表用户的优先级，其范围是 0～3。

4）配置 FTP 用户的目录权限

配置用户的 FTP 目录权限。在系统视图下进行下列配置：

配置 FTP 用户的目录权限 local-user local-user ftp-directory directory

取消 FTP 用户的目录权限 undo local-user local-user ftp-directory

RADIUS 服务器端也可以配置 ftp-directory，可以在 RADIUS 服务器端的属性字典中加入 Ftp-Directory（106）属性，类型为字符串。此属性为华为定义的属性，使用标准属性 Vendor-Specific，华为公司的 Vendor-Id 为 2011，Ftp-Directory 的属性类型（Vendor type）为 4。

5）配置 PPP 用户的回呼认证

（1）配置 PPP 用户的回呼（Callback）属性

可以用以下命令配置 PPP 用户的回呼号码。在系统视图下进行下列配置：

配置用户回呼属性 local-user local-user callback-number callback-number

取消用户回呼属性 undo local-user local-user callback-number

如果 PPP 没有告知 AAA 此用户是回呼用户，AAA 根据此用户是否配置了 callback-number 来决定。如果配置了 callback-number，AAA 告知 PPP 此用户是回呼用户。反之，不是回呼用户。如果 PPP 告知 AAA 此用户是回呼用户，而且此用户配置了 callback-number，AAA 把该值传给 PPP。

（2）配置 PPP 用户的回呼验证属性

可以用以下命令配置 PPP 用户回呼验证属性。

配置用户回呼不认证 local-user local-user callback-nocheck

配置用户回呼认证 undo local-user local-user callback-nocheck

6）配置 ISDN 用户的主叫号码认证

配置了用户的 call-number 就表明要对此用户进行主叫号码的认证。目前仅限于 ISDN 用户。

配置用户主叫号码 local-user local-user call-number call-number [subcall-number]

取消用户主叫号码 undo local-user local-user call-number

5. 配置 RADIUS 服务器

在系统视图和服务器模板视图下，均可以配置 RADIUS 服务器。如果要配置 RADIUS 服务器模板，需要首先创建 RADIUS 服务器模板并进入 RADIUS 服务器模板视图。

RADIUS 服务器配置包括配置服务器地址和监听端口，以及 RADIUS 协议相关的其他属性，包括共享密匙、重传次数、超时重传的时间间隔、为 PPP 用户指定认证服务器、服务器 down 机后的恢复时间等。

1）配置 RADIUS 服务器模板

RADIUS 服务器模板就是一组 RADIUS 服务器。在配置 RADIUS 服务器模板中的服务器之前，需要先创建 RADIUS 服务器模板。在创建 RADIUS 服务器模板时，如果指定的 RADIUS 服务器模板不存在，则增加一个新的 RADIUS 服务器模板，同时进入 radius template 视图。在 radius template 视图下可以对此 RADIUS 服务器模板配置 RADIUS 服务器及其属性。在系统视图下进行下列配置：

配置 RADIUS 服务器模板 radius-server template server-template-name

删除 RADIUS 服务器模板 undo radius-server template server-template-name

2）配置 RADIUS 服务器

可指定 RADIUS 服务器的地址和监听端口号。用户可以配置多个 RADIUS 服务器。系统视图下最多可以配置 15 个 RADIUS 服务器，在一个 Radius template 中最多可以配置 5 个 RADIUS 服务器。每个 Radius template 只能配置一个主 RADIUS 认证服务器、一个主 RADIUS 计费服务器。配置 RADIUS 主服务器时，如果 Radius template 中已经有主 RADIUS 服务器，则将用新配置的 RADIUS 服务器作为主服务器，原主服务器不再作为主服务器。在没有指定主 RADIUS 服务器时，系统将根据配置时间的先后选择使用的 RADIUS 服务器，当一个服务器失效后，系统会自动选择下一个服务器。在配置了主 RADIUS 服务器后，将首选主 RADIUS 服务器实现 AAA。当主 RADIUS 服务器不能正常工作后，使用其他 RADIUS 服务器工作，每隔一定时间，再尝试主 RADIUS 服务器是否可以正常工作，如果发现其可以正常工作则马上恢复使用主 RADIUS 服务器。在系统视图、radius template 视图下进行下列配置：

配置 RADIUS 服务器 IP 地址（或主机名）、认证端口号和计费端口号

radius server { server-name | server-address } [authentication-port port-number] [accounting-port port-number] [auth-primary] [acct-primary]

取消指定的 RADIUS 服务器 undo radius server { server-name | server-address }

认证端口号的默认值为 1812，设置为 0 表示本服务器不作为认证服务器用。

计费端口号的默认值为 1813，设置为 0 表示本服务器不作为计费服务器用。

使用 auth-primary 可配置 RADIUS 服务器为主认证服务器。使用 acct-primary 可配置 RADIUS 服务器为主计费服务器。

3）配置由用户指定 RADIUS 服务器

可以由用户指定 RADIUS 服务器来对用户进行认证。

注意：*该功能要求用户名为 userid@server 形式，其中 userid 为用户名，server 为使用的 RADIUS 服务器，该功能要与接口上的认证方案配合使用。只有在接口上配置的认证方案的第一种方法为 radius 方法时配置才起作用。对用户进行认证时将试图使用由用户指定的 RADIUS 服务器，当此 RADIUS 服务器不能正常工作时再使用其他 RADIUS 服务器进行认证。*

如果要恢复到不由用户指定 RADIUS 服务器，使用命令 rundo radius appoint-

authentication。请在系统视图、radius template 视图下进行下列配置：

配置由用户指定 RADIUS 服务器 radius appoint-authentication [restricted]

禁止由用户指定 RADIUS 服务器 undo radius appoint-authentication [restricted]

restricted 表示仅限使用指定的 RADIUS 服务器对用户进行认证。配置此选项后，当用户指定的 RADIUS 服务器不存在或不能工作时直接拒绝用户请求。

4）配置 RADIUS 密钥用于加密用户口令及生成回应认证符（Response Authenticator）

在系统视图、radius template 视图下进行下列配置：

配置 RADIUS 密钥 radius shared-key key-string

删除 RADIUS 密钥 undo radius shared-key

注意所配密钥要与 RADIUS 服务器中设定的密钥相同。

5）配置 RADIUS 回应超时时间

对于 Radius client（如路由器）发送出去的包，如果需要 Radius server 应答，则需要设置一个超时定时器，在这个设定的时间内如果没有收到 Radiusserver 的应答，则 Radius client 重发此报文。在系统视图、radius template 视图下进行下列配置：

配置 RADIUS 回应超时时间 radius timer response-timeout seconds

恢复默认的 RADIUS 回应超时时间 undo radius timer response-timeout

默认 RADIUS 回应超时时间为 5s。

6）配置 RADIUS 重传次数

当 Radius client（如路由器）向 RADIUS 服务器发出 AAA 请求后，在规定的超时时限内未得到 RADIUS 服务器发回的应答时，客户端会重传 AAA 请求。重传 AAA 请求计数超出规定最大重传次数后，认为该服务器已不能正常工作。请在系统视图、radius template 视图下进行下列配置：

配置 RADIUS 重传次数 radius retry retry-times

恢复 RADIUS 默认重传次数 undo radius retry

默认重传次数为 3。

7）配置 RADIUS 服务器 down 掉后检测恢复时间间隔

Radius client（如路由器）默认的 RADIUS 服务器 down 掉后恢复的默认时间为 5 分钟。请在系统视图、radius template 视图下进行下列配置：

配置 RADIUS 服务器 down 掉后检测恢复时间 radius timer quiet minutes

恢复 RADIUS 服务器默认的检测恢复时间 undo radius timer quiet

8）配置 RADIUS 实时计费包发送间隔时间

用户通过认证后，NAS 以配置的间隔时间向 RADIUS 服务器发送用户的实时计费信息，如果实时计费请求失败，将根据 aaa accounting-scheme optional 命令配置情况对用户进行处理，如果用户配置了 aaa accounting-scheme optional，NAS 将允许用户继续使用网络服务，反之则 NAS 会将用户挂断。默认不使用实时计费。请在系统视图、radius template 视图下进行下列配置：

配置 RADIUS 实时计费包发送间隔时间 radius timer realtime-accounting minutes

不使用实时计费 undo radius timer realtime-accounting

6.2.3　AAA 和 RADIUS 显示与调试

在完成上述配置后，在所有视图下执行 display 命令可以显示配置后 AAA 和 RADIUS 的运行情况，通过查看显示信息认证配置的效果。执行 debugging 命令可对 AAA 和 RADIUS 进行调试。

显示在线用户情况 display aaa user
查看本地用户数据库 display local-user
打开 AAA 事件调试开关 debugging aaa event
关闭 AAA 事件调试开关 undo debugging aaa event
打开 AAA 原语调试开关 debugging aaa primitive
关闭 AAA 原语调试开关 undo debugging aaa primitive
打开 RADIUS 报文调试开关 debugging radius packet
关闭 RADIUS 报文调试开关 undo debugging radius packet

6.2.4　AAA 和 RADIUS 典型配置举例

1．对 PPP 用户采用 RADIUS 服务器进行认证、计费

1）组网需求

RADIUS 服务器 129.7.66.66 作为主认证和计费服务器，RADIUS 服务器 129.7.66.67 作为备用认证服务器和计费服务器，认证端口号默认为 1812，计费端口号默认为 1813。

2）组网图

AAA 和 RADIUS 示例组网图如图 6-6 所示。

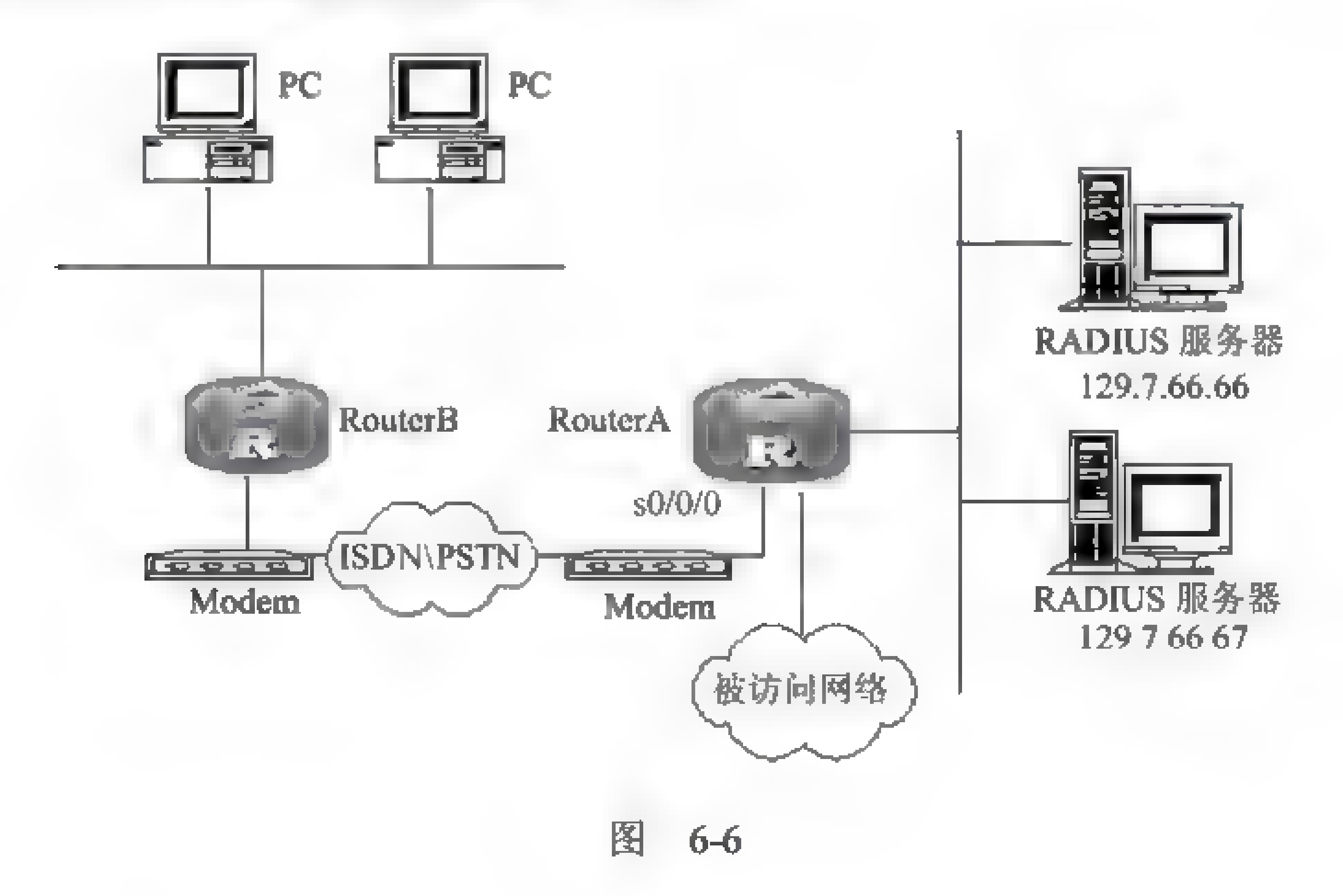


图　6-6

3）配置步骤

配置 RouterA #启动 AAA。

[Quidway] aaa enable

配置 PPP 用户的默认认证方案

[Quidway] aaa authentication-scheme ppp default radius

配置 RADIUS 服务器 IP 地址和端口

[Quidway] radius server 129.7.66.66 auth-primary acct-primary

[Quidway] radius server 129.7.66.67

配置 RADIUS 服务器密钥、重传次数、超时定时器时间长度及计费选项

[Quidway] radius shared-key this-is-my-secret

[Quidway] radius retry 2

[Quidway] aaa accounting-scheme ppp default radius

[Quidway] radius timer response-timeout 5

配置 Serial0/0/0 口应用认证方案

[Router-Serial0/0/0] ppp authentication-mode chap scheme default

2. 对 FTP 用户采用 RADIUS 服务器进行认证

（1）组网需求对 FTP 用户先用 RADIUS 服务器进行认证，如果没有响应，则不认证。认证服务器使用 129.7.66.66，无备用服务器，端口号为默认值 1812。

（2）组网图同上。

（3）配置步骤。

启动 AAA

[Quidway] aaa enable

配置 Login 用户的默认认证方案

[Quidway] aaa authentication-scheme login default radius none

配置 RADIUS 服务器

IP 地址和端口，使用默认端口号。

[Quidway] radius server 129.7.66.66

配置 RADIUS 服务器密钥、重传次数、超时定时器时间长度及 RADIUS 服务器 down 掉后的恢复时间。

[Quidway] radius shared-key this-is-my-secret

[Quidway] radius retry 4

[Quidway] radius timer response-timeout 2

[Quidway] radius timer quiet 1

启动 FTP 服务器

[Quidway] ftp-server enable

6.3 访问控制列表配置

在本小节中主要介绍访问控制列表的概念、创建及详细的配置举例。

6.3.1　访问控制列表简介

访问控制列表（Access Control List，ACL）为网络设备提供基本的服务安全性。对某类服务而言，安全管理员首先应该考虑该服务是否有必要运行在当前环境中。如果有必要，又有哪些用户能够享用该服务。如果该服务不必要，则应当禁止该服务。因为运行这个不必要的服务，不仅会浪费网络等资源，而且会给当前的网络环境带来安全隐患。如果是部分用户需要，则应当为该服务规划权限，禁止无权限的用户使用该服务。如果某类服务仅仅在网络内部需要，则还需尽力避免该服务被网络外部访问。同样，如果某类服务仅在网络外部是必须的，则管理员还应将该服务限制在网络外部。对于某些服务来说，即使用户能使用该服务，安全管理员也应该能监管该服务的使用情况，比如控制某服务只能在某段时间内使用，对该服务的使用量进行统计等。在使用访问控制列表之前，安全管理员必须非常清楚当前网络环境的安全规划和潜在的安全问题。例如在企业网内部，管理员必须清楚部门 A、部门 B 能够访问的服务器内容，部门 A 和部门 B 相互之间能够互通的服务，各部门能够访问的企业网络外部服务，能被企业网络外部访问的服务，以及服务器的访问权限等。对到达端口的数据包进行分类，并打上不同的动作标记，访问列表作用于路由器的所有端口，访问列表的主要用途有包过滤、镜像、流限制、流统计和分配队列优先级等。

当访问控制列表被创建后，既可以以用于拒绝某些数据包经过某个路由器接口，也可用于拒绝某些数据包经过路由器的所有接口。路由器的访问控制列表在创建后将应用于路由器的端口上。默认情况下，路由器将允许所有的数据包经过所有接口，即不做任何转发限制。而采用访问控制列表，则路由器在转发某个数据包之前，将会参考访问控制列表中的内容以确定是否转发。最初，访问控制列表的作用仅限于决定是否转发数据包（如转发或丢弃）。管理员只能对怀疑的数据包作出丢弃决定，但不能监控那些存在安全隐患的数据包。路由器提供给管理员更丰富的功能，如利用访问控制列表进行数据包分析、流量限制和流量统计。使用的技术如下。

- 包过滤（Packet Filtering）技术：指对每个数据包按照用户所定义的项目进行过滤，如比较数据包的源地址、目的地址是否符合规则等。包过滤不涉及会话的状态，也不分析数据，只分析数据包的包头信息。如果配置的规则合理，在这一层能够过滤掉很多有安全隐患的数据包。
- 报文监控（Packet Monitoring）技术：一般 1 台设备需要一个监控端口就可以监控设备的所有端口。监控端口一般接报文/协议分析仪，用于报文/协议分析。管理员事先确定要监控的报文类型，如 ICMP 报文。这时进入或离开路由器的所有 ICMP 报文都会被复制到监控端，连接到该端口的网络分析仪能同时收到该报文进行分析。
- 流量限制（CAR）技术：管理员可以限制某一源与目的对之间的平均报文流量。既可以是 IP 地址对，也可以是 MAC 地址对。流量限制将在两个方向上同时进行。
- 报文统计（Packet Gathering）技术：管理员确定要统计的报文流向，即从何处而来（报文的源地址），准备去哪里（报文的目的地址）。这里的地址既可以是 MAC 地址，也可以是 IP 地址。可以统计双向的报文流量，统计结果即可以是报文的字数，也可以是报文的包数。

下面将主要介绍 ACL 在包过滤方面的应用，实际上 ACL 在 QOS 等有相当广泛的应用，本书不进行详细的介绍。大多数访问控制列表在数据本身上不做任何事，不能基于数据流的实际内容作出操作决定。对报文过滤而言，访问控制列表只能实现不让任何人从外界使用 Telnet 登录。让哪个人用 SMTP 向我们发送电子邮件。哪台机器能把新闻发给我们，但是其他机器不能这样做。然而，访问控制列表不能实现如下的一些需求：某个特定用户能从外部远程登录，但是其他用户不能这样做。因为“用户”不是数据包过滤系统所能辨认的。可以发送这些文件而不是那些文件。“文件”也不是数据包过滤系统所能辨认的。

路由器为了过滤数据包，需要配置一系列的规则，以决定什么样的数据包能够通过，这些规则就是通过访问控制列表 ACL（Access Control List）定义的。访问控制列表是由 permit | deny 语句组成的一系列有顺序的规则，这些规则根据数据包的源地址、目的地址、端口号等来描述。ACL 通过这些规则对数据包进行分类，这些规则应用到路由器接口上，路由器根据这些规则判断哪些数据包可以接收，哪些数据包需要拒绝。

1．访问控制列表的分类

按照访问控制列表的表示方法，可以分为如下两类。

- 名字型的访问控制列表：使用名字来表示一个访问控制列表。
- 数字型的访问控制列表：使用数字来表示一个访问控制列表。

按照访问控制列表的用途，可以分为如下三类。

- 基本的访问控制列表（Basic Acl）。
- 高级的访问控制列表（Advanced Acl）。
- 基于接口的访问控制列表（Interface-based Acl）。

当使用数字型的访问控制列表的时候，访问控制列表的用途是依靠数字的范围来指定的，1～99 范围的数字型访问控制列表是基本的访问控制列表，100～199 范围的数字型访问控制列表是高级的访问控制列表，1000～1999 是基于接口的访问控制列表。当使用名字型的访问控制列表的时候，需要指定该访问控制列表的使用类型。

2．访问控制列表的匹配顺序

现在我们已经介绍了在不同类型的访问控制列表，如何进行规则的配置。由于在一条访问控制列表中，往往会配置多条规则，而每一条规则指定的数据包的范围大小有别，一个访问控制列表可以由多条 permit|deny 语句组成，每一条语句描述的规则是不相同，这些规则可能存在重复或矛盾的地方，在将一个数据包和访问控制列表的规则进行匹配的时候，到底采用哪些规则呢？就需要确定规则的匹配顺序。这样在匹配一个访问控制规则的时候就存在匹配顺序的问题。访问控制列表子规则的匹配顺序有如下两种：

- config：指定匹配该规则时按用户的配置顺序。
- auto：指定匹配该规则时系统自动排序。（按“深度优先”的顺序）

默认情况下匹配顺序为按用户的配置排序，即 config。用户如指定某条访问控制规则的匹配顺序，就不能再更改该顺序，除非把该规则的内容全部删除，再重新指定其匹配顺序。

auto 所用的“深度优先”的原则是指：把指定数据包范围最小的语句排在最前面。这点可以通过比较地址的通配符来实现，通配符越小，则指定的主机的范围就越小。比如

129.102.1.1 0.0.0.0 指定了一台主机：129.102.1.1，而 129.102.1.1 0.0.255.255 则指定了一个网段：从 129.102.1.1 到 129.102.255.255 显然前者在访问控制规则中排在前面。具体标准为：对于基本访问控制规则的语句，直接比较源地址通配符，通配符相同的则按配置顺序；对于基于接 1∶1 过滤的访问控制规则，配置了 any 的规则排在后面，其他按配置顺序；对于高级访问控制规则，首先比较源地址通配符，相同的再比较目的地址通配符，仍相同的则比较端口号的范围，范围小的排在前面，如果端口号范围也相同则按配置顺序，使用 display acl 命令就可以看出是哪条规则首先生效。显示时，列在前面的规则首先生效。

6.3.2　访问控制列表的创建

一个访问控制列表是由 permit | deny 语句组成的一系列的规则列表，若干个规则列表构成一个访问控制列表。在配置访问控制列表的规则之前，首先需要创建一个访问控制列表。

创建一个访问控制列表，先确定 ACL 是数字型的还是名字型的，需要指定如下参数。

ACL 如果是名字型的，ACL 还需要指定 ACL 的用途类型，指定该访问控制列表的匹配顺序，可以使用如下命令创建一个访问控制列表：

acl { number acl-number | name acl-name [basic | advanced | interface] } [match-order { config | auto }]

使用如下的命令删除一个或所有的访问控制列表：

undo acl { number acl-number | name acl-name | all }

参数说明如下。

- number acl-number：定义一个数字型的 ACL。
- name acl-name：定义一个名字型的 ACL。
- basic：定义一个用途类型为基本的 ACL。
- advanced：定义一个用途类型为高级的 ACL。
- interface：定义一个用途类型为基于接口的 ACL。
- acl-number：访问控制规则序号，1～199 和 1000～1999 之间的数字。
- match-order config：指定匹配该规则时按用户的配置顺序。
- match-order auto：指定匹配该规则时系统自动排序，即按“深度优先”的顺序。
- all：删除所有配置的 ACL。

如前所述，默认情况下匹配顺序为按用户的配置排序，即 config。用户一旦指定某一条访问控制列表的匹配顺序，就不能再更改该顺序，除非把该 ACL 的内容全部删除，再重新指定其匹配顺序。

创建了一个访问控制列表之后，将进入 ACL 视图，ACL 视图是按照访问控制列表的用途来分类的，例如创建了一个数字编号为 100 的数字型 ACL，将进入高级 ACL 视图，路由器的提示符如下所示：

[Quidway -acl-adv-100]

如果创建一个名字为 test 的基本 ACL，将进入基本 ACL 视图，路由器的提示符如下：

[Quidway -acl-basic-test]

对于名字型的访问控制列表，在第一次创建的时候，必须指定该访问控制列表的用途类型。创建了命名 ACL 以后，如果再次进入某个命名 ACL 的配置视图，则不需要再指定 ACL 规则的用途类型了，直接指定名字就可以进入相应的配置视图。已经创建的命名 ACL 的用途类型是不能更改的。

创建了一个 ACL，进入了 ACL 视图之后，就可以配置 ACL 的规则了。对于不同的 ACL，其规则是不一样的，具体的各种 ACL 的规则的配置方法将在后面分别介绍。

1. 基本访问控制列表的创建

基本访问控制列表只能使用源地址信息，作为定义访问控制列表的规则的元素。通过上面小节介绍的 acl 的命令，可以创建一个基本的访问控制列表，同时进入基本访问控制列表视图，在基本访问控制列表视图下，可以创建基本访问控制列表的规则。

可以使用如下的命令定义一个基本访问控制列表的规则：

rule [rule-id] { permit | deny } [source sour-addr sour-wildcard | any] [time-range time-name] [logging] [fragment] [vpn-instance vpn-instance-name]

参数说明如下。

- rule-id：可选参数，ACL 规则编号，范围为 0～127。当指定了编号，如果与编号对应的 ACL 规则已经存在，则会使用新定义的规则覆盖旧的定义，相当于编辑一个已经存在的 ACL 的规则。如果与编号对应的 ACL 规则不存在，则使用指定的编号创建一个新的规则。如果不指定编号，表示增加一个新规则，系统自动会为这个 ACL 规则分配一个编号，并增加新规则。
- permit：通过符合条件的数据包。
- deny：丢弃符合条件的数据包。
- source：可选参数，指定 ACL 规则的源地址信息。如果不指定，表示报文的任何源地址都匹配。
- sour-addr：数据包的源地址，点分十进制表示。或用 any 代表源地址 0.0.0.0，通配符 255.255.255.255。
- sour-wildcard：源地址通配符，点分十进制表示。
- time-range：可选参数，指定访问控制列表的生效时间。
- time-name：访问控制列表生效的时间段名字。
- logging：可选参数，是否对符合条件的数据包做日志。日志内容包括访问控制规则的序号，数据包通过或被丢弃，数据包的数目。
- fragment：可选参数，指定该规则是否仅对非首片分片报文有效。当包含此参数时表示该规则仅对非首片分片报文有效。
- vpn-instance：可选参数，指定报文是属于哪个 VPN 实例的。如果没有指定，该规则对所有 VPN 实例中的报文都有效；如果指定了，则表示该规则仅仅对指定的 VPN 实例中的报文有效。

对已经存在的 ACL 规则，如果采用指定 ACL 规则编号的方式进行编辑，没有配置的部分是不受影响的。例如：

先配置了一个 ACL 规则。

rule 1 deny source 1.1.1.1 0

然后再对这个 ACL 规则进行编辑。

rule 1 deny logging

这个时候，ACL 的规则变成如下。

rule 1 deny source 1.1.1.1 0 logging

可以使用如下命令删除一个基本访问控制列表的规则。

undo rule rule-id [source] [time-range] [logging] [fragment] [vpn-instance vpn-instanc-name]

参数说明如下。

- rule-id：ACL 规则编号，必须是一个已经存在的 ACL 规则编号。如果后面不指定参数，则将这个 ACL 规则完全删除。否则只是删除对应 ACL 规则的部分信息。
- source：可选参数，仅仅删除编号对应的 ACL 规则的源地址部分的信息设置。
- time-range：可选参数，仅仅删除编号对应的 ACL 规则在规定时间生效的设置。
- logging：可选参数，仅仅删除编号对应的 ACL 规则对符合条件的数据包做日志的设置。
- fragment：可选参数，仅仅删除编号对应的 ACL 规则仅对非首片分片报文有效的设置。
- vpn-instance：可选参数，仅仅删除编号对应的 ACL 规则中关于 VPN 实例的设置。

2. 高级访问控制列表的创建

高级访问控制列表可以使用数据包的源地址信息、目的地址信息、IP 承载的协议类型、针对协议的特性，例如 TCP 的源端口、目的端口，ICMP 协议的类型、code 等内容定义规则。可以利用高级访问控制列表定义比基本访问控制列表更准确、更丰富、更灵活的规则。

通过前面小节介绍的 acl 的命令，可以创建一个高级的访问控制列表，同时进入高级访问控制列表视图，在高级访问控制列表视图下，可以创建高级访问控制列表的规则。可以使用如下的命令定义一个高级访问控制列表规则：

rule [rule-id] { permit | deny } protocol [source sour-addr sour-wildcard | any] [destination dest-addr dest-mask | any] [soucre-port operator port1 [port2]] [destination-port operator port1 [port2]] [icmp-type icmp-type icmp-code] [precedence precedence] [tos tos] [time-range time-name] [logging] [fragment] [vpn-instance vpn-instance-name]

参数说明如下。

- rule-id：可选参数，ACL 规则编号，范围为 0～127。当指定了编号，如果与编号对应的 ACL 规则已经存在，则会使用新定义的规则覆盖旧的定义，相当于编辑一个已经存在的 ACL 的规则。如果与编号对应的 ACL 规则不存在，则使用指定的编号创建一个新的规则。如果不指定编号，表示增加一个新规则，系统会自动为这个 ACL 规则分配一个编号。
- deny：拒绝符合条件的数据包。
- permit：允许符合条件的数据包。
- protocol：用名字或数字表示的 IP 承载的协议类型。数字范围为 0～255；名字取值范围为 gre、icmp、igmp、ip、ipinip、ospf、tcp 和 udp。

- source：可选参数，指定 ACL 规则的源地址信息。如果不配置，表示报文的任何源地址都匹配。
- sour-addr：数据包的源地址，点分十进制表示；或用 any 代表源地址 0.0.0.0，通配符 255.255.255.255。
- sour-wildcard：源地址通配符，点分十进制表示。
- destination：可选参数，指定 ACL 规则的目的地址信息。如果不配置，表示报文的任何目的地址都匹配。
- dest-addr：数据包的目的地址，点分十进制表示；或用 any 代表目的地址 0.0.0.0，通配符 255.255.255.255。
- dest-wildcard：目的地址通配符，点分十进制表示；或用 any 代表目的地址 0.0.0.0，通配符 255.255.255.255。
- icmp-type：可选参数，指定 ICMP 报文的类型和消息码信息，仅仅在报文协议是 ICMP 的情况下有效。如果不配置，表示任何 ICMP 类型的报文都匹配。
- icmp-type：ICMP 包可以依据 ICMP 的消息类型进行过滤。取值为 0～255 的数字。
- icmp-code：依据 ICMP 的消息类型进行过滤的 ICMP 包也可以依据消息码进行过滤。取值为 0～255 的数字。
- icmp-message：ICMP 包可以依据 ICMP 消息类型名字或 ICMP 消息类型和码的名字进行过滤。
- soure-port：可选参数，指定 UDP 或者 TCP 报文的源端口信息，仅仅在规则指定的协议号是 TCP 或者 UDP 有效。如果不指定，表示 TCP/UDP 报文的任何源端口信息都匹配。
- destination-port：可选参数，指定 UDP 或者 TCP 报文的目的端口信息，仅仅在规则指定的协议号是 TCP 或者 UDP 有效。如果不指定，表示 TCP/UDP 报文的任何目的端口信息都匹配。
- operator：可选参数。比较源或者目的地址的端口号的操作符，名字及意义为 lt（小于）、gt（大于）、eq（等于）、neq（不等于）和 range（在范围内）。只有 range 需要两个端口号做操作数，其他的只需要一个端口号做操作数。
- port1，port2：可选参数。TCP 或 UDP 的端口号，用名字或数字表示，数字的取值范围为 0～65535。
- precedence：可选参数，数据包可以依据优先级字段进行过滤。取值为 0～7 的数字，或名字。
- tos：可选参数，数据包可以依据服务类型字段进行过滤。取值为 0～15 的数字，或名字。
- logging：可选参数，是否对符合条件的数据包做日志。日志内容包括访问控制列表规则的序号，数据包通过或被丢弃，IP 承载的上层协议类型，源/目的地址，源/目的端口号，数据包的数目。
- time-name：这条访问控制列表规则在该时间段内有效。
- fragment：指定该规则是否仅对非首片分片报文有效。当包含此参数时表示该规则仅对非首片分片报文有效。

- vpn-instance：可选参数，指定报文是属于哪个 VPN 实例的。如果没有指定，该规则对所有 VPN 实例中的报文都有效；如果指定了，则表示该规则仅仅对指定的 VPN 实例中的报文有效。

这样，用户通过配置防火墙，添加适当的访问规则，就可以利用包过滤来对通过路由器的 IP 包进行检查，从而过滤掉用户不希望通过路由器的包，起到保护网络安全的作用。

3. 基于接口的访问控制列表的创建

基于接口的访问控制列表，是一种特殊的访问控制列表，可以根据接收报文的接口指定规则。通过前面小节介绍的 acl 的命令，可以创建一个基于接口的访问控制列表，同时进入基于接口的访问控制列表视图，在基于接口的访问控制列表视图下，可以创建基于接口的访问控制列表规则。

可以使用如下的命令定义一个基于接口的访问控制列表规则：

rule { permit | deny } [interface interface-name] [time-range time-name] [logging]

参数说明如下。

- rule-id：可选参数，ACL 规则编号，范围为 0～127。当指定了编号，如果与编号对应的 ACL 规则已经存在，则会使用新定义的规则覆盖旧的定义，相当于编辑一个已经存在的 ACL 的规则。如果与编号对应的 ACL 规则不存在，则使用指定的编号创建一个新的规则。如果不指定编号，表示增加一个新规则，系统自动会为这个 ACL 规则分配一个编号，并增加新规则。
- deny：丢弃符合条件的数据包。
- permit：通过符合条件的数据包。
- interface：可选参数，指定数据包的接口信息。如果不指定，表示所有的接口都匹配。
- interface-name：指定数据包进入的接口名。或者用 any 代表所有的接口。
- logging：可选参数，是否对符合条件的数据包做日志。日志内容包括访问控制列表规则的序号，数据包通过或被丢弃，数据包的数目。
- time-range：可选参数，指定规则在一定时间段内有效。
- time-name：这条访问控制列表规则在该时间段内有效。

可以使用如下的命令删除一个基于接口的访问控制列表的规则：

undo rule rule-id

参数说明：rule-id 为 ACL 规则编号，必须是一个已经存在的 ACL 规则编号。

4. ACL 对分片报文的支持

传统的包过滤并不处理所有 IP 报文分片，而是只对第一个（首片）分片报文进行匹配处理，后续分片一律放行。这样，网络攻击者可能构造后续的分片报文进行流量攻击，就带来了安全隐患。

VRP 平台的包过滤提供了对分片报文过滤的功能，包括：对所有的分片报文进行三层（IP 层）的匹配过滤；同时，对于包含扩展信息的 ACL 规则项（例如包含 TCP/UDP 端口号，ICMP 类型），提供标准匹配和精确匹配两种匹配方式。标准匹配即三层信息的匹配，

匹配将忽略三层以外的信息；精确匹配则对所有的 ACL 项条件进行匹配，这就要求防火墙必须记录首片分片报文的状态以获得完整的后续分片的匹配信息。默认的功能方式为标准匹配方式。

在 ACL 的规则配置项中，通过关键字 fragment 来标识该 ACL 规则仅对非首片分片报文有效，而对非分片报文和首片分片报文则忽略此规则。而不包含此关键字的配置规则项对所有报文均有效。

例如：

[Quidway-basic-1] rule deny source 202.101.1.0 0.0.0.255 fragment

[Quidway-basic-1] rule permit source 202.101.2.0 0.0.0.255

[Quidway-adv-101] rule permit ip destination 171.16.23.1 0 fragment

[Quidway-adv-101] rule deny ip destionation 171.16.23.2 0

上述规则项中，所有项对非首片分片报文均有效；第一，三项对非分片和首片分片报文是被忽略的，仅仅对非首片分片报文有效。

6.3.3 访问控制列表配置举例

某公司通过一台 Quidway 路由器的接口 Serial1 /0/0 访问 Internet，路由器与内部网通过以太网接口 Ethernet0/0/0 连接。公司内部对外提供 WWW. FTP 和 Telnet 服务，公司内部子网为 129.38.1.0。其中，内部 FTP 服务器地址为 129.38.1.1，内部 Telnet 服务器地址为 129.38.1.2，内部 WWW 服务器地址为 129.38.1.3，公司对外地址为 202.38.160.1 在路由器上配置了地址转换，这样内部 PC 可以访问 Internet，外部 PC 可以访问内部服务器。通过配置路由器的防火墙功能，希望实现以下要求：

- 外部网络只有特定用户地址可以访问内部服务器。
- 内部网络只有特定主机可以访问外部网络。

假定外部特定用户的 IP 地址为 202.39.2.3。

1. 组网图

包过滤防火墙（路由器）访问控制列表配置案例组网图如图 6-7 所示。

2. 配置步骤

#在路由器 Q 上允许防火墙

[Q] firewall enable

#设防火墙默认过滤方式为允许包通过

[Q]firewall default permit

#创建访问控制列表 1010

[Q] acl number 101

#配置规则禁止所有 IP 包通过

[Q-acl-adv-101] rule deny ip

#配置规则允许特定主机访问外部网，允许内部服务器访问外部网

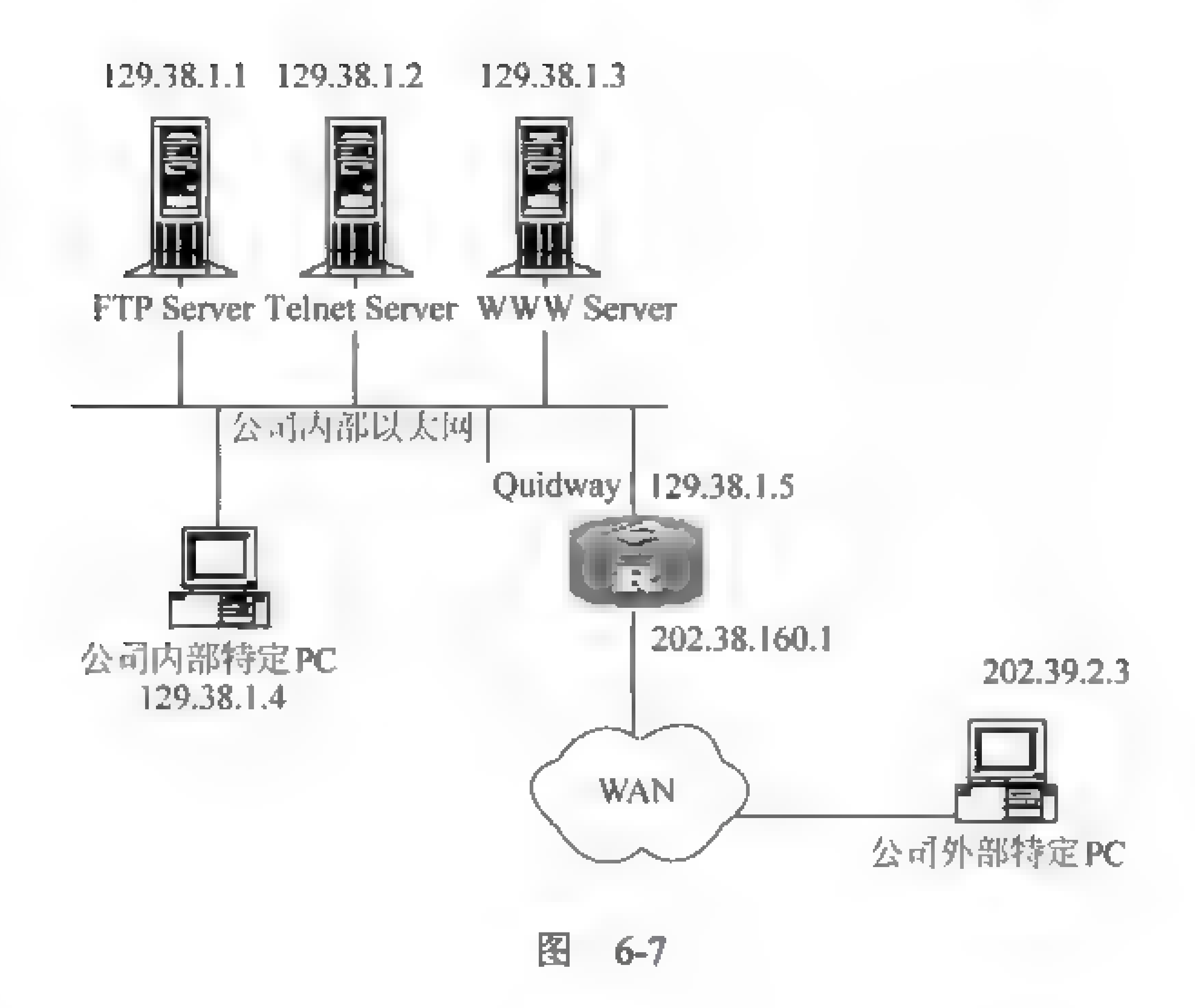


图　6-7

[Q-acl-adv-101] rule permi ip source 129.38.1.4 0
[Q-acl-adv-101] rule permi ip source 129.38.1.1 0
[Q-acl-adv-101] rule permi ip source 129.38.1.2 0
[Q-acl-adv-101] rule permi ip source 129.38.1.3 0
#创建访问控制列表 102
[Q] acl number 102
#配置规则允许特定用户从外部网访问内部服务器
[Q-acl-adv-102]rule permit tcp source 202.39.2.3 0 destination 202.38.160.1 0
#配置规则允许特定用户从外部网取得数据（只允许端口大于 1024 的包）
[Q-acl-adv-102] rule permit tcp destination 202.38.160.10 0 destination-port gt 1024
#将规则 101 作用于从接口 Ethernet0/0/0 进入的包
[Q-Ethernet0/0/0]firewall packet-filter 101 inbound
#将规则 102 作用于从接口 Serial1 /0/0 进入的包
[Q-Serial1/0/0]]firewall packet-filter 102 inbound

6.4　IPSec 与 IKE 技术与配置

在本小节中主要介绍 IPSec 和 IKE 协议及这两个协议的配置与调试，最后给出了 IPSec 的典型配置案例。

6.4.1　IPSec 概述

IPSec（IP Security）协议族是 IETF 制定的一系列协议，它为 IP 数据报提供了高质量的、可互操作的、基于密码学的安全性。特定的通信方之间在 IP 层通过加密与数据源验证

等方式，来保证数据报在网络上传输时的私有性、完整性、真实性和防重放。

- 私有性（confidentiality）：指对用户数据进行加密保护，用密文的形式传送。
- 完整性（Data Integrity）：指对接收的数据进行验证，以判定报文是否被篡改。
- 真实性（Data Authentication）：指验证数据源，以保证数据来自真实的发送者。
- 防重放（Anti-replay）：指防止恶意用户通过重复发送捕获到的数据包所进行的攻击，即接收方会拒绝旧的或重复的数据包。

IPSec 通过 AH（Authentication Header，认证头）和 ESP（Encapsulating Security Payload，封装安全载荷）安全协议来实现上述目标。并且还可以通过 IKE（Internet Key Exchange，因特网密钥交换协议）为 IPSec 提供自动协商交换密钥、建立和维护安全联盟的服务，以简化 IPSec 的使用和管理。

AH 是报文头验证协议，主要提供的功能有数据源验证、数据完整性校验和防报文重放功能；然而，AH 并不加密所保护的数据报。

ESP 是封装安全载荷协议，它除提供 AH 协议的所有功能之外（数据完整性校验不包括 IP 头），还可提供对 IP 报文的加密功能。

AH 和 ESP 可以单独使用，也可以同时使用。对于 AH 和 ESP，都有两种操作模式：传输模式和隧道模式。

IKE 用于协商 AH 和 ESP 所使用的密码算法，并将算法所需的必备密钥放到恰当位置。IKE 协商并不是必须的，IPSec 所使用的策略和算法等也可以手工协商。

6.4.2　IPSec 与 IKE 协议基本概念

1．安全联盟

IPSec 在两个端点之间提供安全通信，端点被称为 IPSec 对等体。IPSec 能够允许系统、网络的用户或管理员控制对等体间安全服务的粒度。例如，某个组织的安全策略可能规定来自特定子网的数据流应同时使用 AH 和 ESP 进行保护，并使用 3DES（Triple Data Encryption Standard，三重数据加密标准）进行加密；另一方面，策略可能规定来自另一个站点的数据流只使用 ESP 保护，并仅使用 DES 加密。通过 SA（Security Association，安全联盟），IPSec 能够对不同的数据流提供不同级别的安全保护。

安全联盟是 IPSec 的基础，也是 IPSec 的本质。SA 是通信对等体间对某些要素的约定，例如，使用哪种协议（是 AH 或 ESP 还是两者结合使用）、协议的操作模式（传输模式和隧道模式）、密码算法（DES 和 3DES）、特定流中保护数据的共享密钥以及密钥的生存周期等。安全联盟是单向的，在两个对等体之间的双向通信，最少需要两个安全联盟来分别对两个方向的数据流进行安全保护。同时，如果希望同时使用 AH 和 ESP 来保护对等体间的数据流，则分别需要两个 SA，一个用于 AH，另一个用于 ESP。

安全联盟由一个三元组来唯一标识，这个三元组包括 SPI（Security Parameter Index，安全参数索引）、目的 IP 地址、安全协议号（AH 或 ESP）。SPI 是为唯一标识 SA 而生成的一个 32 比特的数值，它在 AH 和 ESP 头中传输。

安全联盟具有生存周期。生存周期的计算包括两种方式：

- 以时间为限制：每隔指定长度的时间就进行更新。
- 以流量为限制：每传输指定的数据量（字节）就进行更新。

2．IPSec 协议的操作模式

IPSec 协议有两种操作模式：传输模式和隧道模式。SA 中指定了协议的操作模式。

在传输模式下，AH 或 ESP 被插入到 IP 头之后但在所有传输层协议之前，或所有其他 IPSec 协议之前。在隧道模式下，AH 或 ESP 插在原始 IP 头之前，另外生成一个新头放到 AH 或 ESP 之前。不同安全协议在传输模式和隧道模式下的数据封装形式（传输协议以 TCP 为例）如图 6-8 所示。

模式 / 协议	transport	tunnel
AH	[IP Header] [AH] [TCP Header] [data]	[new IP Header] [AH] [raw IP Header] [TCP Header] [data]
ESP	[IP Header] [ESP] [TCP Header] [data] [ESP Tail] [ESP Auth data]	[new IP Header] [ESP] [raw IP Header] [TCP Header] [data] [ESP Tail] [ESP Auth data]
AH-ESP	[IP Header] [AH] [ESP] [TCP Header] [data] [ESP Tail] [ESP Auth data]	[new IP Header] [AH] [ESP] [raw IP Header] [TCP Header] [data] [ESP Tail] [ESP Auth data]

图　6-8

从安全性来讲，隧道模式优于传输模式。它可以完全地对原始 IP 数据报进行验证和加密；此外，可以使用 IPSec 对等体的 IP 地址来隐藏客户机的 IP 地址。从性能来讲，隧道模式比传输模式占用更多带宽，因为它有一个额外的 IP 头。因此，到底使用哪种模式需要在安全性和性能间进行权衡。

3．验证算法与加密算法

1）验证算法

AH 和 ESP 都能够对 IP 报文的完整性进行验证，以判别报文在传输过程中是否被篡改。验证算法的实现主要是通过杂凑函数，杂凑函数是一种能够接受任意长的消息输入，并产生固定长度输出的算法，该输出称为消息摘要。IPSec 对等体计算摘要，如果两个摘要是相同的，则表示报文是完整未经篡改的。一般来说 IPSec 使用两种验证算法。

- MD5：通过输入任意长度的消息，产生 128bit 的消息摘要。
- SHA-1：通过输入长度小于 2^{64}b 的消息，产生 160b 的消息摘要。SHA-1 的摘要长于 MD5，因而是更安全的。

2）加密算法

ESP 能够对 IP 报文内容进行加密保护，防止报文内容在传输过程中被窥探。加密算法实现主要通过对称密钥系统，它使用相同的密钥对数据进行加密和解密。VRP 中 IPSec 实现三种加密算法。

- DES（Data Encryption Standard）：使用 56b 的密钥对一个 64b 的明文块进行加密。

- 3DES（Triple DES）：使用三个 56b 的 DES 密钥（共 168b 密钥）对明文进行加密。
- AES（Advanced Encryption Standard）：VRP 实现了 128b 密钥长度的 AES 算法，这也是 IETF 标准要求实现的。

4．协商方式

有两种协商方式建立安全联盟，一种是手工方式（manual），一种是 IKE 自动协商（isakmp）方式。前者配置比较复杂，创建安全联盟所需的全部信息都必须手工配置，而且 IPSec 的一些高级特性（例如定时更新密钥）不被支持，但优点是可以不依赖 IKE 而单独实现 IPSec 功能。而后者则相对比较简单，只需要配置好 IKE 协商安全策略的信息，由 IKE 自动协商来创建和维护安全联盟。当与之进行通信的对等体设备数量较少时，或是在不动静态环境中，手工配置安全联盟是可行的。对于中、大型的动态网络环境中，推荐使用 IKE 协商建立安全联盟。

在实施 IPSec 的过程中，可以使用因特网密钥交换 IKE（Internet Key Exchange）协议来建立安全联盟，该协议建立在由 Internet 安全联盟和密钥管理协议 ISAKMP（Internet Security Association and Key Management Protocol）定义的框架上。IKE 为 IPSec 提供了自动协商交换密钥、建立安全联盟的服务，能够简化 IPSec 的使用和管理。

如前所述，网络安全包括两层含义：其一是内部网的安全，其二是在公共网络中进行数据交换的安全。实现前者的手段有防火墙、地址转换（NAT）等。后者如正在兴起的 IPSec（IP Security），IPSec 提供了在 IP 层对报文实施加密的保护手段。IPSec 的安全联盟可以通过手工配置的方式建立，但是当网络中结点增多时，手工配置将非常困难，而且难以保证安全性。这时就要使用 IKE 自动地进行安全联盟建立与密钥交换的过程。IKE 具有一套自保护机制，可以在不安全的网络上安全地分发密钥、验证身份、IPSec 安全联盟。

IKE 的安全机制包括：

- DH（Diffie-Hellman）交换及密钥分发：该算法是一种公共密钥算法。通信双方在不传送密钥的情况下通过交换一些数据，计算出共享的密钥。加密的前提是交换加密数据的双方必须要有共享的密钥。IKE 的精髓在于它永远不在不安全的网络上直接传送密钥，而是通过一系列数据的交换，最终计算出双方共享的密钥。即使第三者（如黑客）截获了双方用于计算密钥的所有交换数据，也不足以计算出真正的密钥。
- 完善的前向安全性（Perfect Forward Secrecy，PFS）：PFS 特性是一种安全特性，指一个密钥被破解，并不影响其他密钥的安全性，因为这些密钥间没有派生关系。对于 IPsec，是通过在 IKE 第二阶段协商中增加一次密钥交换并由 DH 算法保障的。
- 身份验证：身份验证确认通信双方的身份。对于 pre-shared key 验证方法，验证字用来作为一个输入产生密钥，验证字不同是不可能在双方产生相同的密钥的。验证字是验证双方身份的关键。

身份保护，身份数据在密钥产生之后加密传送，实现了对身份数据的保护。IKE 使用了两个阶段为 IPSec 进行密钥协商并建立安全联盟：第一阶段，通信各方彼此间建立了一个已通过身份验证和安全保护的通道，此阶段的交换建立了一个 ISAKMP 安全联盟，即 ISAKMP SA；第二阶段，用在第一阶段建立的安全通道为 IPSec 协商安全服务，即为 IPSec

协商具体的安全联盟，建立 IPSec SA，IPSec SA 用于最终的 IP 数据安全传送。

从图 6-9 中可以看出 IKE 和 IPSec 的关系。

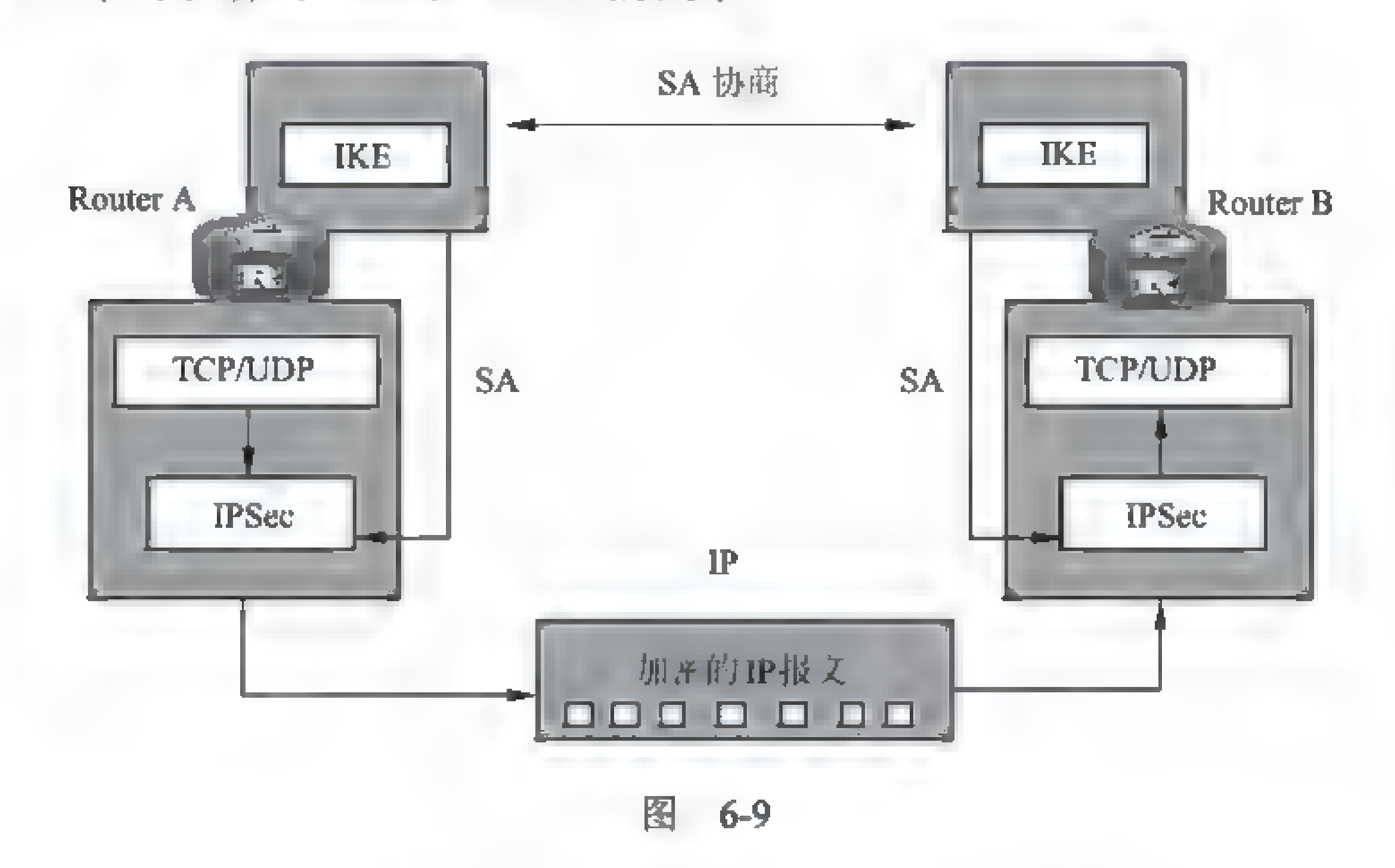


图　6-9

另外，为了使 IKE 支持目前广泛应用的通过 ADSL 及拨号方式构建 VPN 的方案中的特殊情况——局端设备的 IP 地址为固定分配的，用户端设备的 IP 地址为动态获取的情况，在 IKE 阶段的协商模式中增加了 IKE 野蛮模式，它可以选择根据协商发起端的 IP 地址或者 ID 来查找对应的身份验证字，并最终完成协商。IKE 野蛮模式相对于主模式来说更加灵活，能够支持协商发起端为动态 IP 地址的情况。

在 IPSec/IKE 组建的 VPN 隧道中，若存在 NAT 网关设备，且 NAT 网关设备对 VPN 业务数据流进行了 NAT 转换的话，则必须配置 IPSec/IKE 的 NAT 穿越功能。该功能删去了 IKE 协商过程中对端 UDP 端口号的验证过程，同时实现了对 VPN 隧道中 NAT 网关设备的发现功能，即如果发现 NAT 网关设备，则将在之后的 IPSec 数据传输中使用 UDP 封装（即将 IPSec 报文封装到 IKE 协商所使用的 UDP 连接隧道里）的方法，避免了 NAT 网关对 IPSec 报文进行篡改（NAT 网关设备将只能够修改最外层的 IP 和 UDP 报文头，对 UDP 报文封装的 IPSec 报文将不作修改），从而保证了 IPSec 报文的完整性（IPSec 数据加密解密验证过程中要求报文原封不动地被传送到接收端）。

6.4.3　IPSec 在 VRP 上的配置与实现方法

IPSec 实现方式是基于下列思路：通过 IPSec，对等体之间（此处是指 VRP 所在路由器及其对端）能够对不同的数据流实施不同的安全保护（验证、加密或两者同时使用）。其中数据流的区分通过配置 ACL 来进行；安全保护所用到的安全协议、验证算法和加密算法、操作模式等通过配置安全提议来进行；数据流和安全提议的关联（即定义对何种数据流实施何种保护）、SA 的协商方式、对等体 IP 地址的设置（即保护路径的起/终点）、所需要的密钥和 SA 的生存周期等通过配置安全策略来进行；最后在路由器接口上实施安全策略即完成了 IPSec 的配置。主要步骤如下：

1）定义被保护的数据流

数据流是一组流量（traffic）的集合，由源地址/掩码、目的地址/掩码、IP 报文承载的协议号、源端口号和目的端口号等来规定。一个数据流用一个 ACL 来定义，所有匹配一个访问控制列表规则的流量，在逻辑上作为一个数据流。一个数据流可以小到是两台主机之间单一的 TCP 连接；也可以大到是两个子网之间所有的流量。IPSec 能够对不同的数据流施加不同的安全保护，因此 IPSec 配置的第一步就是定义数据流。

2）定义安全提议

安全提议规定了对要保护的数据流所采用的安全协议、验证或加密算法、操作模式（即报文的封装方式）等。VRP 支持的 AH 和 ESP 安全协议，两者既可单独使用，也可联合使用。其中，AH 支持 MD5 和 SHA-1 验证算法； ESP 协议支持 MD5、SHA-1 验证算法和 DES、3DES 加密算法。

VRP 支持的操作模式包括传输模式和隧道模式。对同一数据流，对等体两端必须设置相同的协议、算法和操作模式。另外，对于两个安全网关（例如 VRP 路由器间）实施 IPSec，建议采用隧道模式，以隐藏实际通信的源和目的 IP 地址。因此，请先根据需要配置好一个安全提议，以便下一步将数据流和安全提议相关联。

3）定义安全策略或安全策略组

安全策略规定了对什么样的数据流采用什么样的安全提议。一条安全策略由“名字”和“顺序号”共同唯一确定。安全策略分为手工安全策略和 IKE 协商安全策略，前者需要用户手工配置密钥、SPI 和 SA 的生存周期等参数，在隧道模式下还需要手工配置安全隧道两个端点的 IP 地址；后者则由 IKE 自动协商生成这些参数。安全策略组是所有具有相同名字、不同顺序号的安全策略的集合。在同一个安全策略组中，顺序号越小的安全策略，优先级越高。

4）接口实施安全策略

在接口上应用安全策略组，安全策略组中的所有安全策略同时应用在这个接口上，从而实现对流经这个接口的不同的数据流进行不同的安全保护。

根据上面的介绍，IPSec 主要配置如下。

（1）配置访问控制列表。

（2）定义安全提议，它包括创建安全提议、选择安全协议、选择安全算法和选择报文封装形式。

（3）创建安全策略包括手工创建安全策略和用 IKE 创建安全策略。其中手工创建安全策略包括在安全策略中引用安全提议、在安全策略中引用访问控制列表、配置隧道的起点和终点、配置安全联盟的 SPI、配置安全联盟使用的密钥。用 IKE 创建安全策略包括在安全策略中引用安全提议、在安全策略中引用访问控制列表、在安全策略中引用 IKE 对等体、配置安全联盟生存周期（可选）和配置协商时使用的 PFS 特性。

（4）配置安全策略模板（可选）。

（5）在接口上应用安全策略。

下面介绍具体的配置过程。

1. 定义访问控制列表

用于 IPSec 的访问控制列表的作用不同于在防火墙中所介绍的访问控制列表。一般的

访问控制列表是用来决定一个接口上哪些数据可通过，哪些要被拒绝；而 IPSec 是根据访问控制列表中的规则来确定哪些报文需要安全保护，哪些报文不需要安全保护，故用于 IPSec 的访问控制列表可称为加密访问控制列表。加密访问控制列表匹配（permit）的报文将被保护，加密访问控制列表拒绝（deny）的报文将不被保护。加密访问控制列表既可用于加密入口数据流，也可用于加密出口数据流。在本地和远端路由器上定义的加密访问控制列表必须是相对应的（即互为镜像），这样在某一端加密的数据才能在对端上被解密。例如：

本端：

acl number 101

rule 1 permit ip source 173.1.1.1 0.255.255.255 destination 173.2.2.2 0.255.255.255

对端：

acl number 101

rule 1 permit ip source 173.2.2.2 0.255.255.255 destination 173.1.1.1 0.255.255.255

IPSec 对访问控制列表（ACL）中 permit 的数据流进行保护，因此建议用户精确地配置 ACL，只对确实需要 IPSec 保护的数据流配置 permit，避免盲目地使用关键字 any。建议用户将本端和对端的 ACL 配置成互为镜像。

当用户使用 display acl 命令来浏览路由器的访问控制列表，所有扩展 IP 访问控制列表都将显示在命令的输出中，即同时包括了用于通信过滤和用于加密的扩展 IP 访问控制列表，该命令的输出信息不区分这两种不同用途的扩展访问控制列表。

2. 定义安全提议

安全提议保存 IPSec 需要使用的特定安全性协议以及加密/验证算法，为 IPSec 协商安全联盟提供各种安全参数。为了能够成功的协商 IPSec 的安全联盟，两端必须使用相同的安全提议。

安全提议的配置包括：

- 定义安全提议。
- 选择安全协议。
- 选择安全算法。
- 设置安全协议对 IP 报文的封装模式。

1）创建安全提议

安全提议是用于实施 IPSec 保护而采用的安全协议、算法和报文封装形式的一个组合。一条安全策略通过引用一个或多个安全提议来确定采用的安全协议、算法和报文封装形式。在安全策略引用一个安全提议之前，这个安全提议必须已经建立。最多能够创建 50 个安全提议。

可对安全提议进行修改，但对已协商成功的安全联盟，新修改的安全提议并不起作用，即安全联盟仍然使用原来的安全提议（除非使用 reset ipsec sa 命令重置），只有新协商的安全联盟将使用新的安全提议。

在系统视图下进行下列配置：

创建安全提议并进入安全提议视图 ipsec proposal proposal-name。

删除安全提议 undo ipsec proposal proposal-name。

2）选择报文封装形式

在安全提议中需要指定报文封装模式，安全隧道的两端所选择的 IP 报文封装模式必须一致。在安全提议视图下进行下列配置：

设置安全协议对 IP 报文的封装形式 encapsulation-mode { transport | tunnel }。

恢复默认报文封装形式 undo encapsulation-mode。

通常，在两个安全网关（路由器）之间，总是使用隧道模式。而在两台主机之间的通讯，或者是一台主机和一个安全网关之间的通讯（例如网关工作站和一台路由器之间的网关通讯，此时安全网关相对于网关数据来说是接收主机）选择传输模式。默认情况下采用 tunnel，即隧道模式。

3）选择安全协议

安全提议中需要选择所采用的安全协议。目前可选的安全协议有 AH 和 ESP，也可指定同时使用 AH 与 ESP。安全隧道两端所选择的安全协议必须一致。请在安全提议（IPsec proposal）视图下进行下列配置：

设置安全提议采用的安全协议 transform { ah | ah-esp | esp }。

恢复默认的安全协议 undo transform。

默认情况下采用 esp，即 RFC2406 规定的 ESP 协议。

4）选择安全算法

不同的安全协议可以采用不同的验证算法和加密算法。目前，AH 支持 MD5 和 SHA-1 验证算法；ESP 协议支持 MD5、SHA-1 验证算法和 DES、3DES、AES 加密算法。请在安全提议视图下进行下列配置：

设置 ESP 协议采用的加密算法 esp encryption-algorithm { 3des | des | aes }。

设置 ESP 协议不对报文进行加密 undo esp encryption-algorithm。

设置 ESP 协议采用的验证算法 esp authentication-algorithm { md5 | sha1 }。

设置 ESP 协议不对报文进行验证 undo esp authentication-algorithm。

设置 AH 协议采用的验证算法 ah authentication-algorithm { md5 | sha1 }。

恢复 AH 协议默认的验证算法 undo ah authentication-algorithm。

ESP 协议允许对报文同时进行加密和验证，或只加密，或只验证。注意，undo esp authentication-algorithm 命令不是恢复验证算法为默认算法，而是设置验证算法为空，即不验证。当加密算法为空时，undo esp authentication-algorithm 命令失效。AH 协议没有加密的功能，只对报文进行验证。undo ah authentication-algorithm 命令用来恢复 AH 协议默认验证算法（md5）。在安全隧道的两端设置的安全策略所引用的安全提议必须设置成采用同样的验证算法和/或加密算法。默认情况下，ESP 协议采用的加密算法是 des，采用的验证算法是 md5；AH 协议采用的验证算法是 md5。注意必须首先通过 transform 命令选择了相应的安全协议后，该安全协议所需的安全算法才可配置。例如，如果使用 transform 命令选择了 esp，那么只有 ESP 所需的安全算法才可配置，而 AH 所需的安全算法则不能配置。

3. 创建安全策略

安全策略规定了对什么样的数据流采用什么样的安全提议。安全策略分为手工安全策

略和 IKE 协商安全策略，前者需要用户手工配置密钥、SPI 等参数，在隧道模式下还需要手工配置安全隧道两个端点的 IP 地址；后者则由 IKE 自动协商生成这些参数。

本文将全面介绍安全策略的各项配置，以 IKE 协商创建安全策略联盟的配置为例介绍用 IKE 创建安全策略联盟方法。

IKE 创建安全策略联盟的配置包括：

- 用 IKE 创建安全策略联盟。
- 配置安全策略引用的访问控制列表。
- 指定安全隧道的终点。
- 配置安全策略中引用的安全提议。
- 配置安全联盟的生存时间。

1）用 IKE 创建安全策略

在系统视图下进行下列配置：

用 IKE 创建安全策略，进入安全策略视图。

ipsec policy policy-name seq-number isakmp

用 IKE 并采用策略模板动态创建安全策略。

ipsec policy policy-name seq-number isakmp [template template-name]

修改通过 IKE 协商建立的安全策略。

ipsec policy policy-name seq-number isakmp

删除安全策略。

undo ipsec policy policy-name [seq-number]

2）配置在安全策略中引用安全提议

安全策略通过引用安全提议来确定采用的安全协议、算法和报文封装形式。在引用一个安全提议之前，这个安全提议必须已经建立。请在安全策略视图下进行下列配置：

设置安全策略所引用的安全提议 proposal proposal-name1 [proposal-name2 … proposal-name6]。

取消安全策略引用的安全提议 undo proposal。

通过手工（manual）方式建立安全联盟，一条安全策略只能引用一个安全提议，并且如果已经设置了安全提议，必须先取消原先的安全提议才能设置新的安全提议。在安全隧道的两端设置的安全策略所引用的安全提议必须设置成采用同样的安全协议、算法和报文封装形式。

3）配置在安全策略引用的访问控制列表

安全策略引用访问控制列表，IPSec 根据该访问控制列表中的规则来确定哪些报文需要安全保护，哪些报文不需要安全保护：访问控制列表匹配（permit）的报文被保护，访问控制列表拒绝（deny）的报文不被保护。命令如下：

设置安全策略引用的访问控制列表 security acl acl-number。

取消安全策略引用的访问控制列表 undo security acl。

一条安全策略只能引用一条访问控制列表，如果设置安全策略引用了多个访问控制列表，只有最后配置的那条访问控制列表才能有效通过 IKE（isakmp）协商建立安全联盟，一条安全策略最多可以引用 6 个安全提议，IKE 协商将在安全隧道的两端搜索能够完全匹

配的安全提议。如果IKE在两端找不到完全匹配的安全提议，则安全联盟不能建立，需要被保护的报文将被丢弃。

4）配置在安全策略中引用IKE对等体

对于IKE协商方式，无需像手工方式那样配置对等体、SPI和密钥等参数，IKE将自动协商它们，因而仅需要将安全策略和IKE Peer关联即可。命令如下：

在安全策略中引用 ike对等体 ike peer peer-name。

取消在安全策略中引用ike对等体 undo ike peer peer-name。

IPSec对IKE Peer的引用，实际在IKE Peer视图下还需要进行一些IKE相关参数的设置，包括IKE的协商模式、ID类型、NAT穿越、共享密钥、对端地址和对端名称等。

5）配置安全联盟的生存周期（可选）

（1）配置全局的安全联盟生存周期

所有在安全策略视图下没有单独配置生存周期的安全联盟，都采用全局生存周期。IKE为IPSec协商建立安全联盟时，采用本地设置的和对端提议的生存周期中较小的一个。有两种类型的生存周期："基于时间"的生存周期和"基于流量"的生存周期。无论哪一种类型的生存周期先到期，安全联盟都会失效。安全联盟快要失效前，IKE将为IPSec协商建立新的安全联盟，这样在旧的安全联盟失效时新的安全联盟就已经准备好。在系统视图下进行下列配置：

设置全局的安全联盟（SA）生存周期。

ipsec sa global-duration { traffic-based kilobytes | time-based seconds }

恢复全局的安全联盟（SA）生存周期为默认值 undo ipsec sa global-duration { traffic-based | time-based }。

改变全局生存周期，不会影响单独配置了自己的生存周期的安全策略，也不会对已经建立的安全联盟产生影响，但是在以后的IKE协商中会用于建立新的安全联盟。生存周期只对通过 isakmp方式建立的安全联盟有效，对通过manual方式建立的安全联盟没有生存周期的限制，即手工建立的安全联盟永远不会失效。

（2）配置安全联盟的生存周期

为安全策略设置单独的安全联盟生存周期，如果没有单独设置生存周期，则采用设定的全局生存周期。IKE为IPSec协商建立安全联盟时，采用本地设置的和对端提议的生存周期中较小的一个。请在安全策略视图下进行下列配置：

设置安全策略安全联盟的生存周期 sa duration { traffic-based kilobytes | time-based seconds }。

恢复使用设定的全局生存周期 undo sa duration { traffic-based | time-based }。

改变生存周期，不会影响已经建立的安全联盟，但是在以后的IKE协商中会用于建立新的安全联盟。

6）配置协商时使用的PFS特性（可选）

PFS（Perfect Forward Secrecy，完善的前向安全性）是一种安全特性，指一个密钥被破解，并不影响其他密钥的安全性，因为这些密钥间没有派生关系。此特性是通过在IKE阶段的协商中增加密钥交换来实现的。请在安全策略视图下进行下列配置：

设置协商时使用的PFS特性 pfs { dh-group1 | dh-group2 }。

设置在协商时不使用 PFS 特性 undo pfs。

IKE 在使用此安全策略发起一个协商时，进行一个 PFS 交换。如果本端指定了 PFS，对端在发起协商时必须是 PFS 交换。本端和对端指定的 DH 组必须一致，否则协商会失败。

1024-bit Diffie-Hellman 组（group2）比 768-bit Diffie-Hellman 组（group1）提供更高的安全性，但是需要更长的计算时间。

4．配置安全策略模板

在采用 IKE 方式创建安全策略时，除直接在安全策略视图下直接配置安全策略外，还可以通过引用安全策略模板来创建安全策略。在这种情况下，我们应先在安全策略模板中配置好所有的安全策略。安全策略模板的配置与普通的安全策略配置相似，首先创建一个策略模板，然后配置模板的参数。

创建/修改 IPsec 安全策略模板 ipsec policy-template template-name seq-number。

删除安全策略模板 undo ipsec policy-template policy-template-name [seq-number]。

执行上面的创建命令，会进入 IPsec 策略模板视图，在此视图下，可以配置策略模板的参数。

安全策略模板可配置的参数与 IPsec 安全策略相同，只是很多参数是可选的。必须配置的参数只有 IPsec 安全提议，而隧道对端地址、保护的数据流、PFS 特性可以不配置。但需要注意：如果配置了这些参数中的一个或几个，则在协商时这些参数必须匹配。

在策略模板配置完成后，还需要使用如下命令引用 IPsec 安全策略模板 ipsec policy policy-name seq-number template template-name。

当某一个安全策略引用了安全策略模板后，就不能够再进入其安全策略视图下配置或修改安全策略了，只能进入安全策略模板视图下配置或修改。

5．在接口上应用安全策略组

为使定义的安全联盟生效，应在每个要加密的出站数据、解密的入站数据所在接口（逻辑的或物理的）上应用一个安全策略组，由这个接口根据所配置的安全策略组和对端加密路由器配合进行报文的加密处理。当安全策略组被从接口上删除后，此接口便不再具有 IPSec 的安全保护功能。请在接口视图下进行下列配置：

应用安全策略组 ipsec policy policy-name。

取消应用的安全策略组 undo ipsec policy。

一个接口只能应用一个安全策略组，一个安全策略组可以应用到多个接口上。但手工方式配置的安全策略只能应用到一个接口。当从一个接口发送报文时，将按照从小到大的顺序号查找安全策略组中每一条安全策略。如果报文匹配了一条安全策略引用的访问控制列表，则使用这条安全策略对报文进行处理；如果报文没有匹配安全策略引用的访问控制列表，则继续查找下一条安全策略；如果报文对所有安全策略引用的访问控制列表都不匹配，则报文直接被发送（IPSec 不对报文加以保护）。

华为公司实现的 IPSec 安全策略除了可以应用到串口、以太网口等实际物理接口上之外，还能够应用到 Tunnel、Virtual Template 等虚接口上。这样就可以根据实际组网要求，

在如 GRE、L2TP 等隧道上应用 IPSec。

6.4.4 IPSec 显示与调试

IPSec 提供以下命令显示安全联盟、安全联盟生存周期、安全提议、安全策略的信息以及 IPSec 处理的报文的统计信息。

display 命令可在所有视图下进行操作，debugging 命令只能在用户视图下操作。

显示安全联盟的相关信息 display ipsec sa [brief | remote ip-address | policy policy-name [seq-number] | duration]。

显示 IPSec 处理报文的统计信息 display ipsec statistics。

显示安全提议的信息 display ipsec proposal [proposal-name]。

显示安全策略的信息 display ipsec policy [brief | name policy-name [seq-number]]。

显示安全策略模板的信息 display ipsec policy-template [brief | name policy-name [seq-number]]。

打开 IPSec 的调试功能 debugging ipsec { sa | packet [policy policy-name [seq-number] | parameters ip-address protocol spi-number] | misc }。

禁止 IPSec 的调试功能 undo debugging ipsec { sa | packet [policy policy-name [seq-number] | parameters ip-address protocol spi-number] | misc }。

清除 IPSec 的报文统计信息，此配置任务清除 IPSec 的报文统计信息，所有的统计信息都被设置成零。在用户视图下进行下列操作：

清除 IPSec 的报文统计信息 reset ipsec statistics。

删除安全联盟，此配置任务删除已经建立的安全联盟（无论是手工建立的还是通过 IKE 协商建立的）。如果未指定参数，则删除所有的安全联盟。在用户视图下进行下列操作：

删除安全联盟 reset ipsec sa [remote ip-address | policy policy-name [seq-number] parameters dest-address protocol spi]。

对于通过 IKE 协商建立的安全联盟，被删除后如果有报文重新触发 IKE 协商，IKE 将重新协商建立安全联盟。对于手工建立的安全联盟，被删除后系统会根据手工设置的参数立即创建新的安全联盟。如果指定参数 parameters，由于安全联盟是成对出现的，删除了一个方向安全联盟，另一个方向安全联盟也随之被删除。

6.4.5 IPSec 典型配置案例

1. 采用 isakmp 方式建立安全联盟的配置组网需求

在 Router A 和 Router B 之间建立一个安全隧道，对 PC A 代表的子网（10.1.1.x）与 PC B 代表的子网（10.1.2.x）之间的数据流进行安全保护。安全协议采用 ESP 协议，加密算法采用 DES，验证算法采用 SHA1-HMAC-96。

2. 组网图

IPSec 配置组网图如图 6-10 所示。

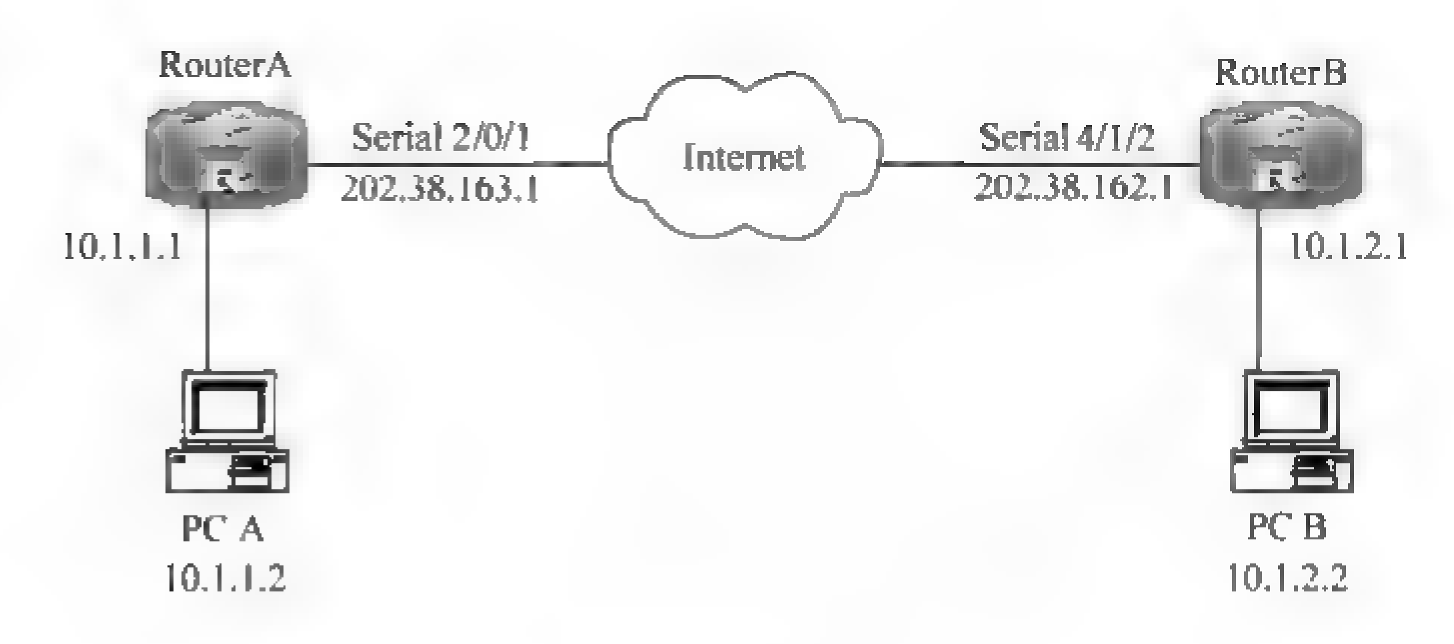


图 6-10

3. 配置步骤

1）配置 Router A

配置一个访问控制列表，定义由子网 10.1.1.x 去子网 10.1.2.x 的数据流

[Quidway] acl number 101

[Quidway-acl-adv-101] rule permit ip source 10.1.1.0 0.0.0.255 destination 10.1.2.0 0.0.0.255

[Quidway-acl-adv-101] rule deny ip source any destination any

配置到 PC B 的静态路由

[Quidway] ip route-static 10.1.2.0 255.255.255.0 202.38.162.1

创建名为 tran1 的安全提议

[Quidway] ipsec proposal tran1

报文封装形式采用隧道模式

[Quidway-ipsec-proposal-tran1] encapsulation-mode tunnel

安全协议采用 ESP 协议

[Quidway-ipsec-proposal-tran1] transform esp

选择算法

[Quidway-ipsec-proposal-tran1] esp encryption-algorithm des

[Quidway-ipsec-proposal-tran1] esp authentication-algorithm sha1

退回到系统视图

[Quidway-ipsec-proposal-tran1] quit

配置 IKE 对等体

[Quidway] ike peer peer

[Quidway-ike-peer-peer] pre-share-key abcde

[Quidway-ike-peer-peer] remote- address 202.38.162.1

创建一条安全策略，协商方式为 isakmp

[Quidway] ipsec policy map1 10 isakmp

引用安全提议

[Quidway-ipsec-policy-isakmp-map1-10] proposal tran1

引用访问控制列表

[Quidway-ipsec-policy-isakmp-map1-10] security acl 101

引用 IKE 对等体

[Quidway-ipsec-policy-isakmp-map1-10] ike peer peer

退回到系统视图

[Quidway-ipsec-policy-isakmp-map1-10] quit

进入串口配置视图

[Quidway] interface serial 12/0/1

配置串口的 IP 地址

[Quidway-Serial12/0/1] ip address 202.38.163.1 255.0.0.0

在串口上应用安全策略组

[Quidway-Serial12/0/1] ipsec policy map1

退回到系统视图

[Quidway-Serial12/0/1] quit

2）配置 Router B

配置一个访问控制列表，定义由子网 10.1.2.x 去子网 10.1.1.x 的数据流

[Quidway] acl number 101

[Quidway-acl-adv-101] rule permit ip source 10.1.2.0 0.0.0.255 destination 10.1.1.0 0.0.0.255

[Quidway-acl-adv-101] rule deny ip source any destination any

配置到 PC A 的静态路由

[Quidway] ip route-static 10.1.1.0 255.255.255.0 202.38.163.1

创建名为 tran1 的安全提议

[Quidway] ipsec proposal tran1

报文封装形式采用隧道模式

[Quidway-ipsec-proposal-tran1] encapsulation-mode tunnel

安全协议采用 ESP 协议

[Quidway-ipsec-proposal-tran1] transform esp

选择算法

[Quidway-ipsec-proposal-tran1] esp encryption-algorithm des

[Quidway-ipsec-proposal-tran1] esp authentication-algorithm sha1

退回到系统视图

[Quidway-ipsec-proposal-tran1] quit

配置 IKE 对等体

[Quidway] ike peer peer

[Quidway-ike-peer-peer] pre-share-key abcde

[Quidway-ike-peer-peer] remote-address 202.38.163.1

创建一条安全策略，协商方式为 isakmp

[Quidway] ipsec policy use1 10 isakmp

```
# 引用访问控制列表
[Quidway-ipsec-policy-isakmp-use1-10] security acl 101
# 引用安全提议
[Quidway-ipsec-policy-isakmp-use1-10] proposal tran1
# 引用 IKE 对等体
[Quidway-ipsec-policy-isakmp-map1-10] ike peer peer
# 退回到系统视图
[Quidway-ipsec-policy-isakmp-use1-10] quit
# 进入串口配置视图
[Quidway] interface serial 4/1/2
# 配置串口的 IP 地址
[Quidway-Serial4/1/2] ip address 202.38.162.1 255.0.0.0
# 在串口上应用安全策略组
[Quidway-Serial4/1/2] ipsec policy use1
# 退回到系统视图
[Quidway-Serial4/1/2] quit
```

以上配置完成后，Router A 和 Router B 之间如果有子网 10.1.1.x 与子网 10.1.2.x 之间的报文通过，将触发 IKE 进行协商建立安全联盟。IKE 协商成功并创建了安全联盟后，子网 10.1.1.x 与子网 10.1.2.x 之间的数据流将被加密传输。

第7章 电子邮件的安全管理

现在的 Internet 上，使用最广泛的应用服务之一就是电子邮件服务，上网的人每人几乎至少有一个邮箱，E-mail（电子邮件）是一个时髦的词，每个人都离不开电子邮件，因为它已经成为互联时代必不可少的产物，它使天南海北的人距离拉近，很多的信息交流在几秒内就可以在网上传递，在电子邮箱中还会有许多个人的隐私。因此在本章中详细讲述电子邮件系统的安全知识，并且讲述对电子邮件服务器 Exchange 的安全配置。

7.1 电子邮件概述

电子邮件是一种利用电子手段提供信息交换的通信方式，是 Internet 应用服务中用户最多、使用最广泛的一类服务。当前电子邮件系统提供了进行复杂通信和交互服务的功能，主要是接发电子邮件和对邮件做各种处理。这就是电子邮件的一个子系统——用户代理。

电子邮件的另一个子系统是消息传输代理，顾名思义就是传送邮件消息。电子邮件传递与其他应用服务的不同之处在于，当接收端网络出现故障时邮件系统还必须提供服务，而其他服务则可能重发几次后终止。在电子邮件系统中采用了缓存技术，当用户发送一个邮件消息时，系统将邮件副本给发送者，目的机的标识及时间放入独有的存储区，然后使用后台进程完成邮件的发送。

后台的进程用域名系统将目的机器映射为 IP 地址，然后建立 TCP 连接。连接成功后把报文的副本传递给目的主机，当目的主机发回已收到报文认可后，在缓存中删除副本；如果建立连接不成功，后台进程将尝试在几天内发送，如果仍没传送成功，将给邮件发送者发送失败报告。

7.2 电子邮件使用的协议

电子邮件相关的协议有三种：POP 邮局协议、IMAP 交互式电子邮件访问协议和 SMTP 简单邮件传输协议，下面分别进行介绍。

7.2.1　POP 邮局协议

POP（Post Office Protocol）邮局协议是个说明 PC 如何与 Internet 上的邮件服务器连接及如何下载 E-mail 的协议。POP 邮局协议负责将邮件通过 SLIP/PPP 连接传送到用户的主机上，它是一种只负责接收的协议，不能通过它发送邮件。目前流行的版本是 POP3 协议，它是一种从远程邮箱中读取电子邮件的简单协议。

7.2.2　IMAP 交互式电子邮件访问协议

IMAP（Internet Message Access Protocol）协议是指从公司的邮件服务器获得 E-mail 的有关信息或直接收取邮件的协议。这个 E-mail 协议可以让用户远程拨号连接 Internet 服务器，并且可以在下载邮件之前预览邮件的主题与来源，还可以选择是否下载附件，可以是邮件的一部分或是邮件整体。换言之，就是电子邮件服务器维护一个中心数据库，多台用户计算机能同时访问使用这台邮件服务器。

7.2.3　SMTP 简单电子邮件传输协议

SMTP（Simple Mail Transfer Protocol）是一个简单的 ASCII 协议，它用于接受连接，并将消息发送到目的邮箱，如果传送失败，则返回出错报告。通过 SMTP 协议所指定的服务器，就可以把 E-mail 寄到收信人的服务器上。

目前，多数的电子邮箱都是基于 SMTP 简单电子邮件传输协议和 POP3 邮局协议。支持 IMAP 交互式电子邮件访问协议的很少，21CN 是少数之一。

7.3　电子邮件发送方式的安全

电子邮件的收发方式有如下两种。

- 使用 Web 页方式：用 IE 登录到主页，进入自己的邮箱收发自己的邮件。
- 使用邮件客户端：如使用 OutLook、FoxMail 等。

7.3.1　Web 页方式

WWW 服务是最广泛的使用 Internet 方式，大部分人的邮箱都是基于某一个网站上的 Web 电子邮箱。当登录到站点主页，可以通过身份验证与密码验证的组合进入已经申请的邮箱，如果发送的邮件没有商业或个人隐私时，你或许不太关心它的安全问题，但如有机密，那么你必须知道，最近浏览过的网页会自动保存在缓存文件夹中，并且你进入信箱时的地址栏中的地址也会被保存下来，成为历史记录，所以为了保证你以 Web 方式收发电子邮件的安全，必须打开 IE，并选择工具菜单中的 Internet 选项，这时将出现如图 7-1 所示的“Internet 选项”对话框，在“Internet 临时文件”和“历史记录”两个选项框中，必须

选择删除文件和清除历史记录这两项内容，才能保证收发电子邮件的内容及地址不被窃取。

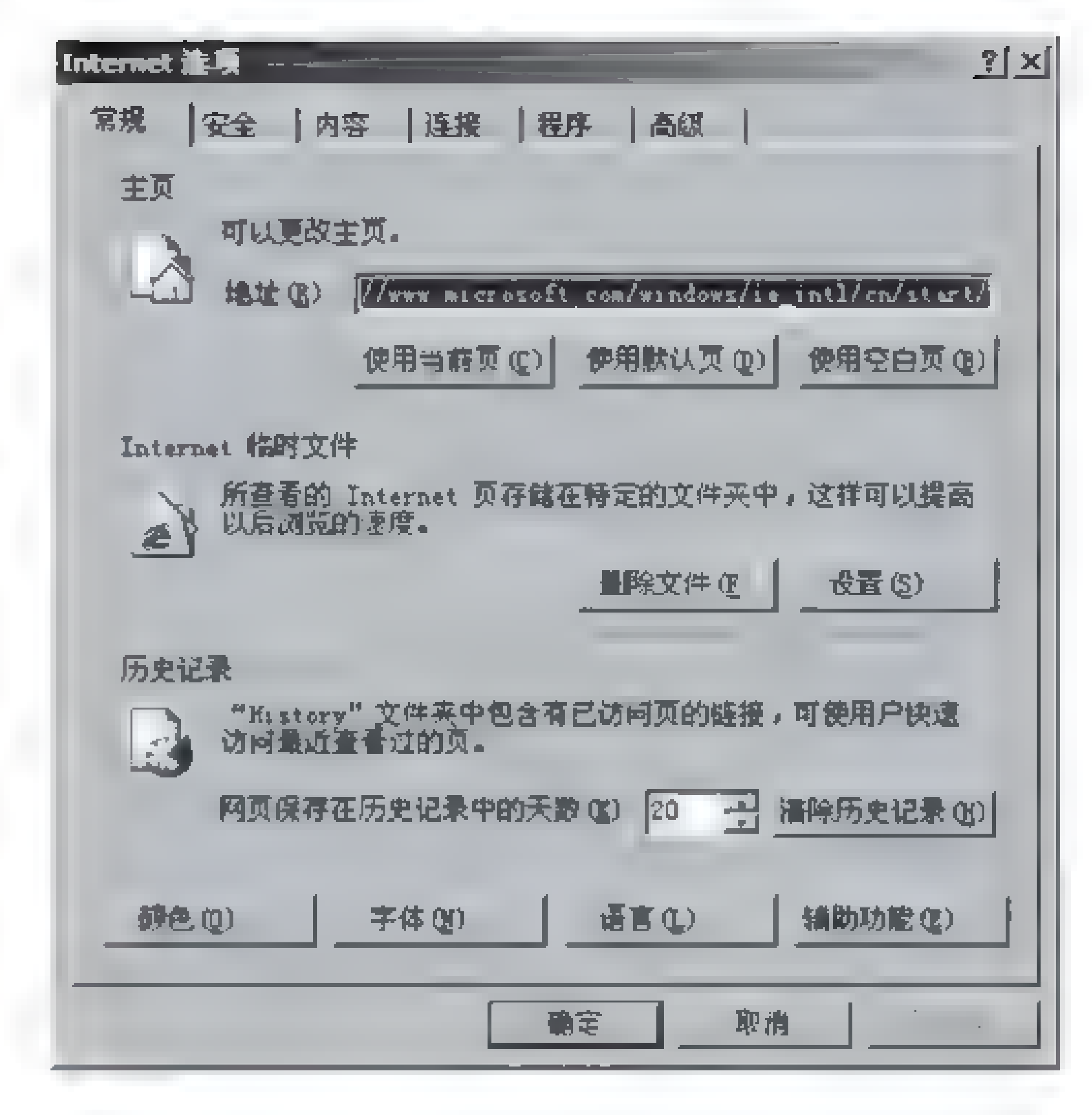


图 7-1

另外还可以在一些安全站点中申请电子邮箱，例如在 www.hotmail.com 站点上申请电子邮箱，Hotmail 站点本身是一个用 SSL 建立的安全站点，当登录到 MSN Hotmail 时，用户的登录名和密码会被加密，并且接下来的数据通过 SSL 连接来进行数据传输，这样就确保了数据在网上传送是在一条安全通道中进行，没有人能窃取在该连接中所传送的数据。另外，当登录并离开安全连接后，MSN Hotmail 还将使用计算机生成的密钥而不是用户登录的密钥进行跟踪，另外还将定期更新计算机生成的密钥以防止其他人员冒充。同时 MSN Hotmail 有垃圾过滤器及自动病毒扫描功能等各种安全方面的设置，解除在 Web 页上收发电子邮件的安全顾虑。通过登录 Hotmail 主页申请邮箱，如图 7-2 所示。

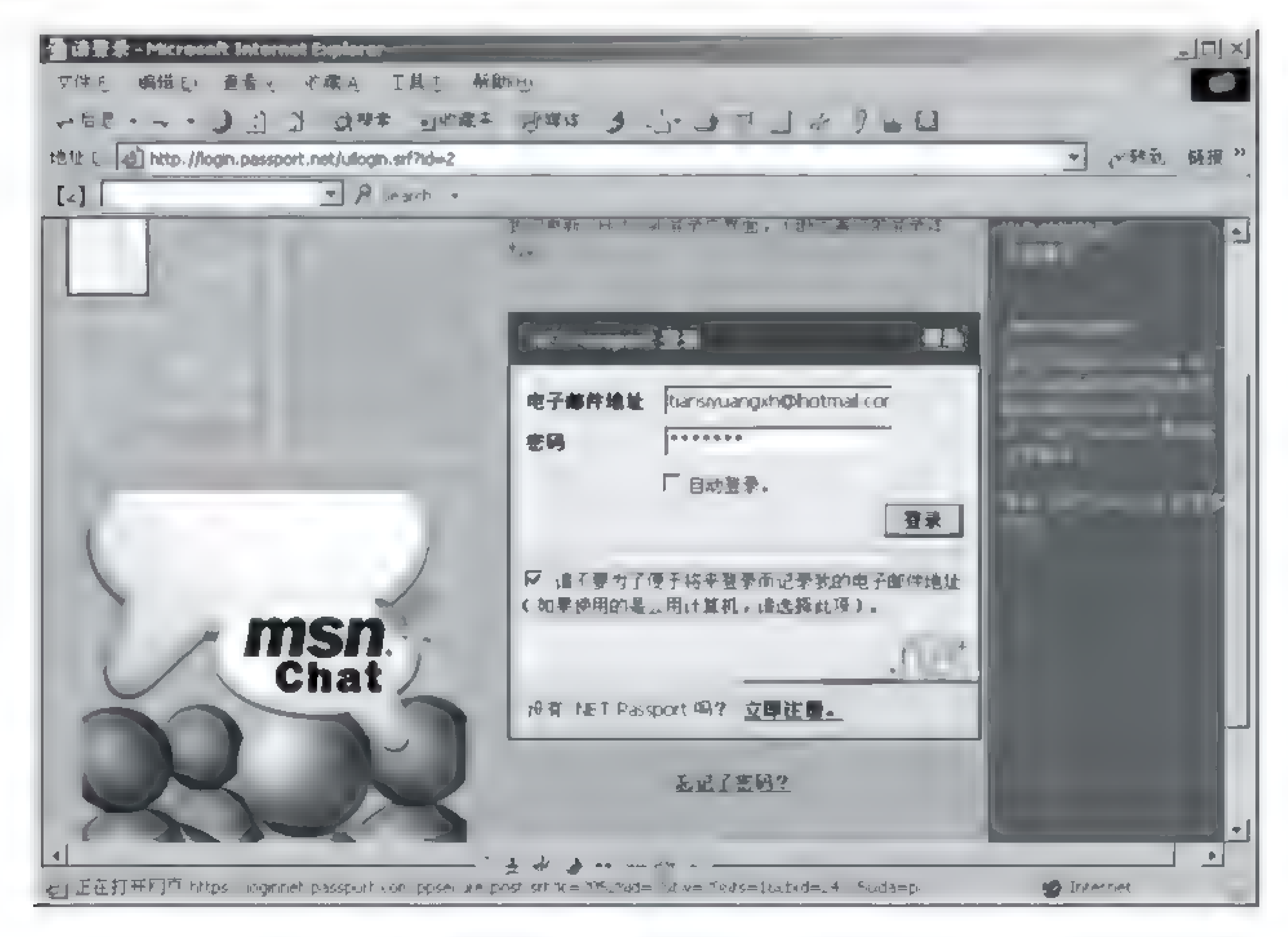


图 7-2

如果使用的是公用计算机，请选择“请不要为了便于将来登录而记录我的电子邮件地址”复选框。

另外，在退出时，请记住在退出时的窗口中单击 Hotmail 页面右上方的“退出”按钮，这样可以防止其他人员使用你的 Passport，如图 7-3 所示。

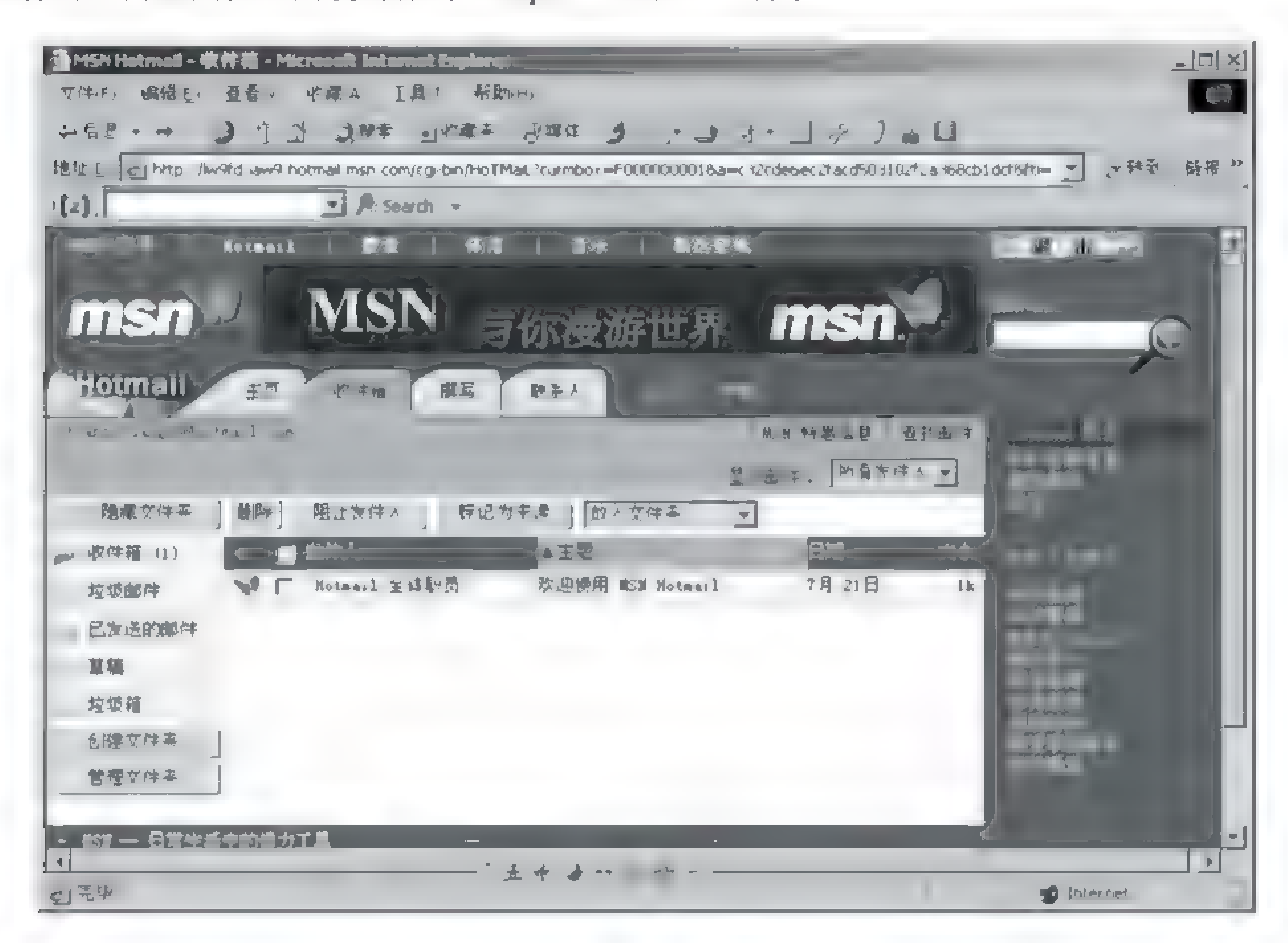


图　7-3

7.3.2　客户端收发电子邮件的安全

相对而言，使用客户端比在 Web 页上收发电子邮件安全一些，常用的有 OutLook、FoxMail 等，下面以 OutLook 为例讲述客户端收发电子邮件的安全。

1．配置 OutLook 的安全区域

OutLook 允许设置安全区域，通过配置安全区域，来增强电子邮件防止非授权访问。具体的操作如下：

打开 OutLook，选择“工具”→“选项”→“安全”选项，出现图 7-4 所示的对话框。在这个对话框中可以单击“区域设置”按钮

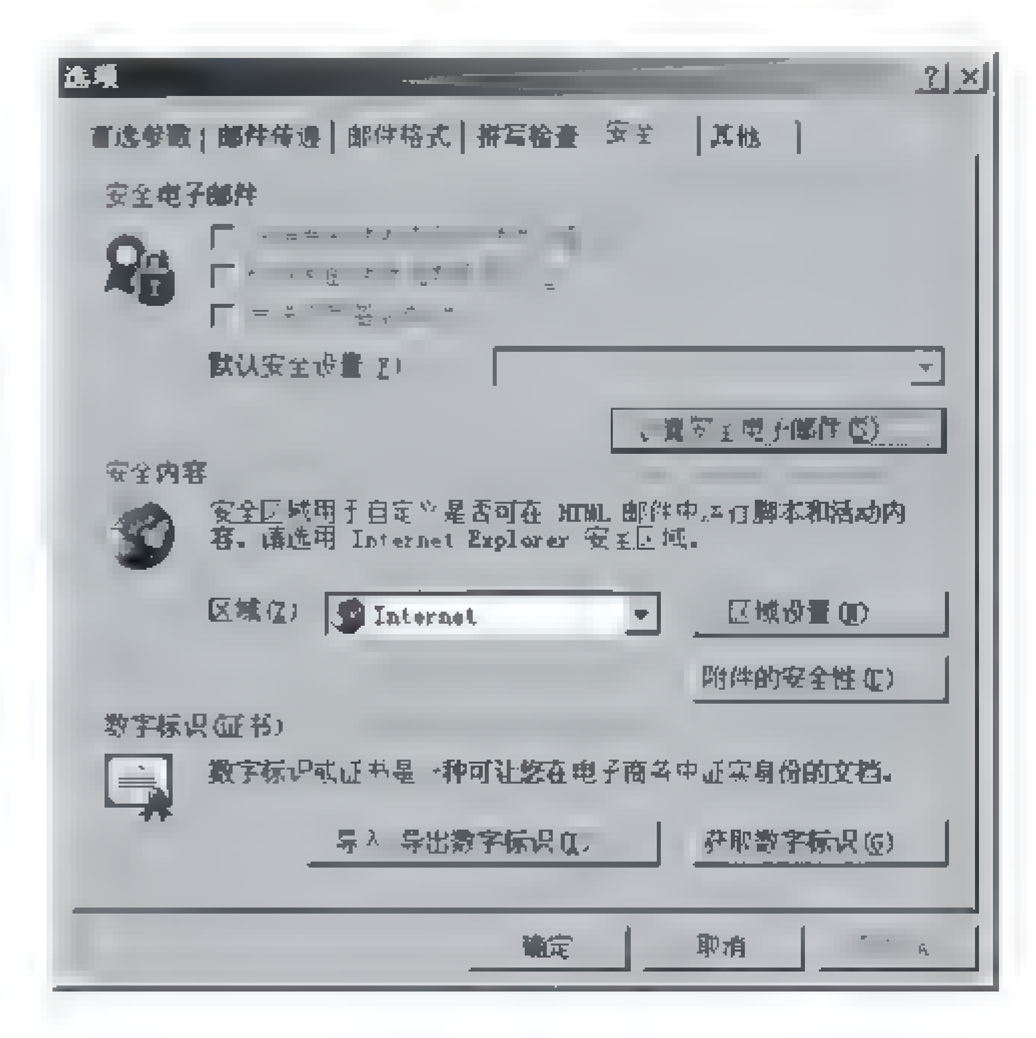


图　7-4

来自定义是否可在 HTML 邮件中运行脚本和活动内容。在所选择的区域中可以有三种安全的级别：高、中、低。

2. 数字标识（数字证书）

为了使 OutLook 能安全地收发电子邮件，必须首先拥有数字标识，因为数字标识就是你在网络上的身份证，只有它，才能安全收发电子邮件。

1）获得数字标识

获得数字标识有两种方法，一种是通过一些受信任的 CA 发放中心来获得数字标识(即数字证）如 Verisign 公司，它的网址是 www.verisign.com，访问此站点，按要求填入正确的个人信息，此公司会给你一个数字身份识别号，然后访问 http://digitalid.verisign.com/mspickup.htm 网页，提交你的数字身份识别号即可。这个数字就是你的数字标识。第二种是通过微软证书服务器来完成获得数字标识（参见第 9 章）。

2）配置数字标识

配置数字标识时，选择“工具”→“账号”选项，选择想使用数字标识的账号，选择“属性”→“安全”选项，再单击“签名标识”区域的“选择”按钮，选择使用该账号签署邮件时将使用何种数字标识；再单击“加密首选项”区域的“选择”按钮，选择加密证书和算法，这些信息将包含在你的数字签名的邮件中，这样，阅读到你的电子邮件的人就可以用同样的设置向你发送加密邮件。

3）备份数字标识

为了防止数字标识在计算机上的丢失，应该对证书进行备份，双击 IE 浏览器，单击“工具”菜单，选择“证书”菜单中的“证书”按钮进入“证书管理器”，选择需要备份的证书（数字标识），单击“导入/导出数字标识”按钮备份数字标识。

3. OutLook 安全电子邮件的配置

有了数字签名后，就可以用 OutLook 发送安全的电子邮件了，从图 7-4 可知，安全的电子邮件有三个选项：

- 将待发邮件的内容和附件加密。
- 给待发邮件添加数字签名。
- 发送文字签名邮件。

选择三项中的任何组合均可达到安全发送电子邮件的目的，如果选择第二项，则所有签名的邮件都会出现一个签名图标，这个图标显示在主题一栏的右下角。

在图 7-4 中单击“设置安全电子邮件”按钮，将出现如图 7-5 所示“更改安全设置”对话框。

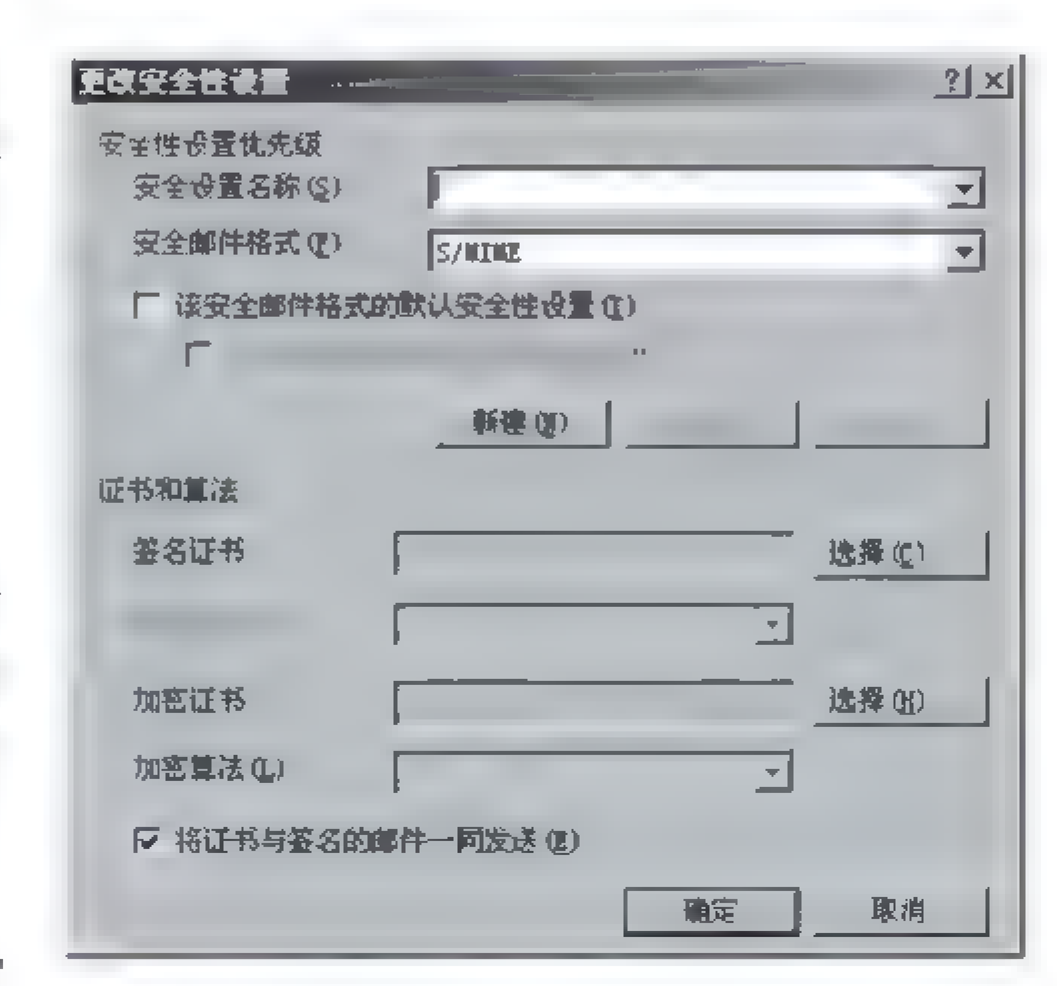


图 7-5

在这个对话框中，选择数字签名证书或加密证书就可以对邮件进行加密或数字签名了，还可以自己选择加密算法，完成后单击“确

定”按钮。

7.4　电子邮件加密工具

电子邮件加密的工具包很多，如 A-Lock、Puffer、PGP 等。

7.4.1　A-Lock 邮件加密软件

A-Lock 是一个邮件加密的免费软件，可以到华军软件园去下载，具体的网址是 http://www.onlinedown.net/alock.htm，并在计算机上安装，安装完成后，在任务上会看到此软件的图标。下面介绍此加密软件的使用。

1．设置密码

右击任务栏中 A-Lock 图标，弹出菜单如图 7-6 所示，选择 View/Edit Password Book 选项，可以为不同的电子邮件设置密码。当然，首先软件必须是已经注册了的。

2．在邮件中使用 A-Lock 进行邮件加密

完成电子邮件设置之后，选中加密内容，然后在图 7-6 中选择 Encrypt/Decrypt 选项，在弹出的对话框中设定密码即可，这时会在加密邮件的头和尾出现 <<START_PC_Encrypt_DATA>> 及 <<END_PC_Encrypt_DATA>>表明加密成功，然后邮件就可以发送出去了。

3．在邮件中使用 A-Lock 进行邮件解密

收到用此加密的电子邮件后，需要选中要解密的邮件内容，再次选择 Encrypt/Decrypt 选项，此软件会自动搜索密码进行解密，如果没有则手工输入密码。

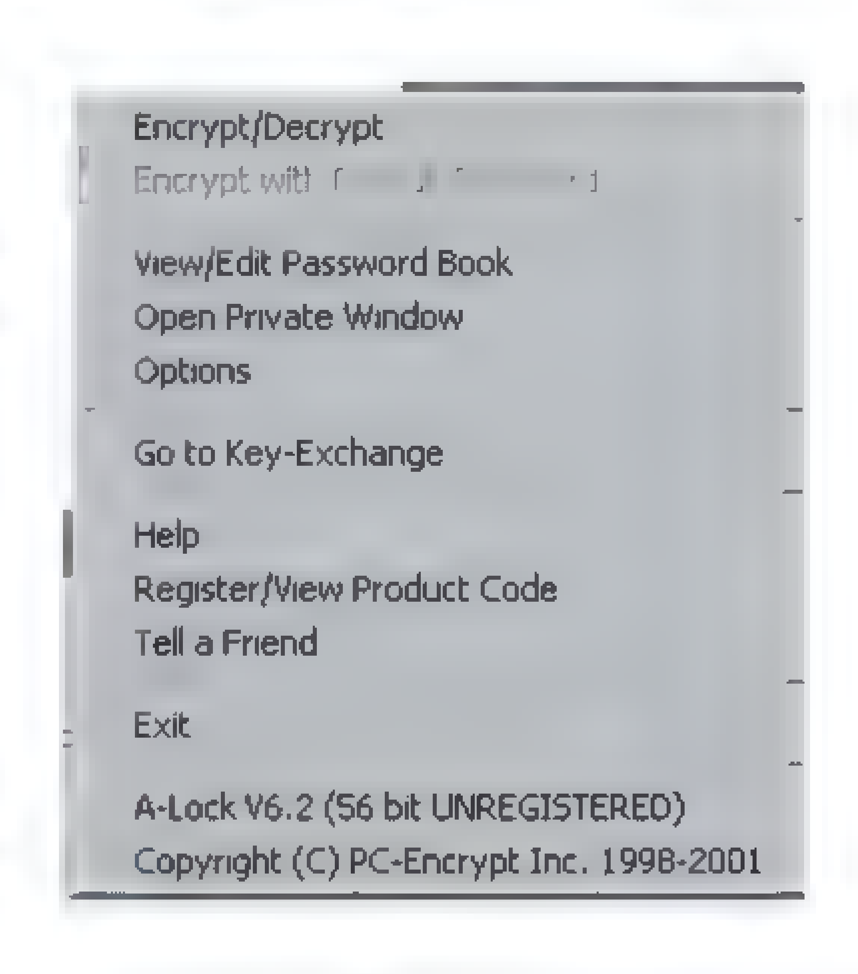


图　7-6

7.4.2　Puffer 邮件加密工具

同样，Puffer 也是一个免费的共享软件，可以到 http://www.briggsoft.com 上进行下载。下载安装后，打开 Puffer3 邮件加密工具如图 7-7 所示。

Puffer3 的加密功能很强大，它使用当前通信领域内最先进的方法，不用别人的密码，只要记住自己的密码就可以实现安全的加密和解密，并且可以加密任何邮件系统和任何文件格式的邮件。从图 7-7 中可知，有如下 6 个选项卡。

- Main：设置 Puffer3.12 内部参数。
- Encrypt：在加密窗口中加密文件。



图 7-7

- Decrypt：在解密窗口中解密文件。
- Wipe：擦除窗口中的文件，也就是删除文件。
- Keys：在钥匙窗口中添加和删除公用钥匙。
- Editor：编辑电子邮件窗口。

所有的操作都是由这 6 个选项卡来完成，对电子邮件的加密，Puffer 提供了两种加密方法：Password（口令）与 PublicKey（公钥）。

1．Password 方法

1）Password 加密

用此方法进行加密，具体的操作步骤如下。

（1）打开 Puffer，单击 Encrypt 菜单，弹出图 7-8 所示的对话框，在 Source 选项区域的单选按钮中选择要加密的文件，这三个选项的说明如下。

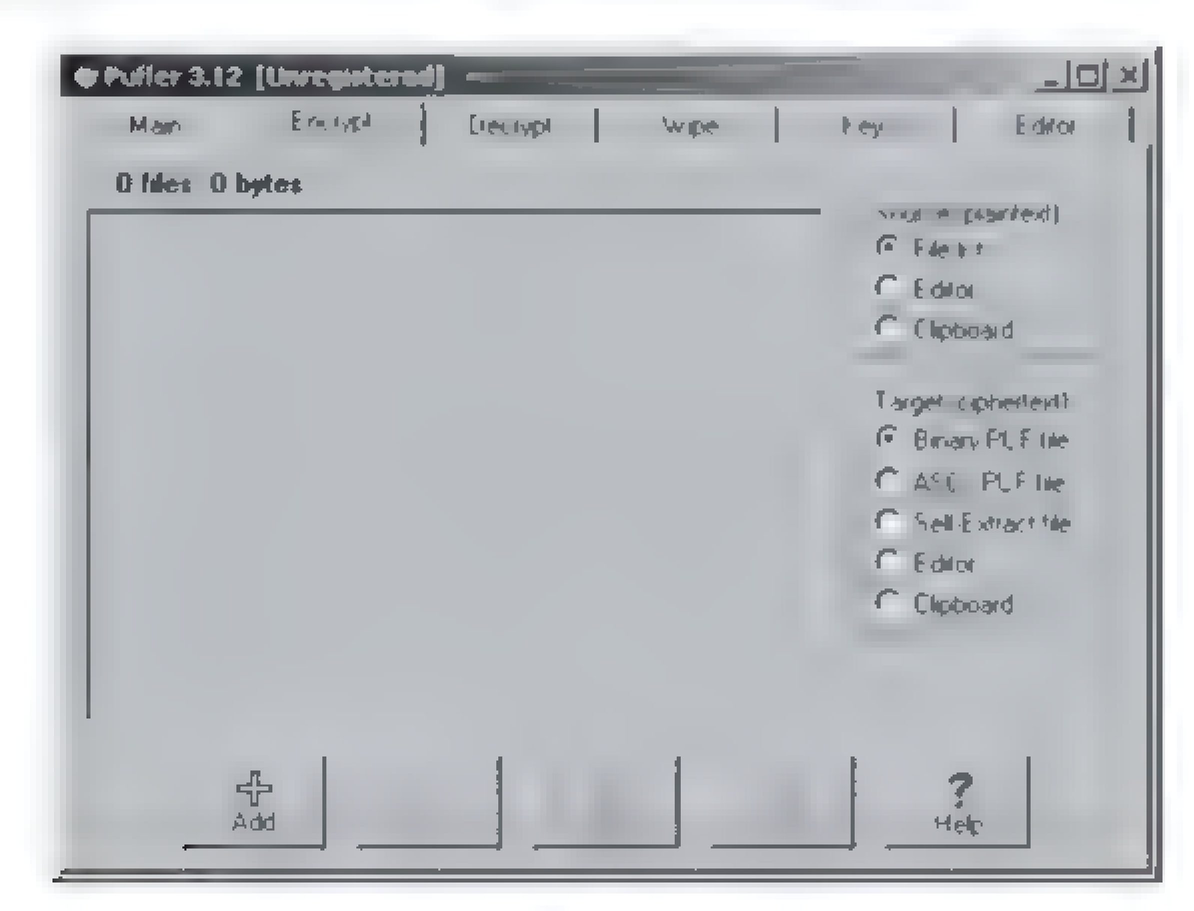


图 7-8

- File list：从硬盘中列出文件。
- Editor：提供编辑器上当前所编写的内容。
- Clipboard：加密当前剪贴板上的内容。

在 Target 选项区域中是文件加密后的形式，单选按钮中各项说明如下。

- Binary PUF file：目标文件被保存为扩展名为.puf 的文件。
- ASCII PUF file：目标文件被保存为 ASCII 形式的扩展名为.puf 的文件。
- Self-Extract file：自动解压缩文件，收信人只要运行该文件，输入正确的口令。
- Editor：目标文件到编辑器上。
- Clipboard：目标文件到剪贴板上。

一般选择默认项，即 Binary PUF file。

（2）选择默认项后单击 Add 按钮，打开图 7-9 所示的对话框，找到要加密的文件，单击 OK 按钮。出现已加上加密文件的对话框如图 7-10 所示。

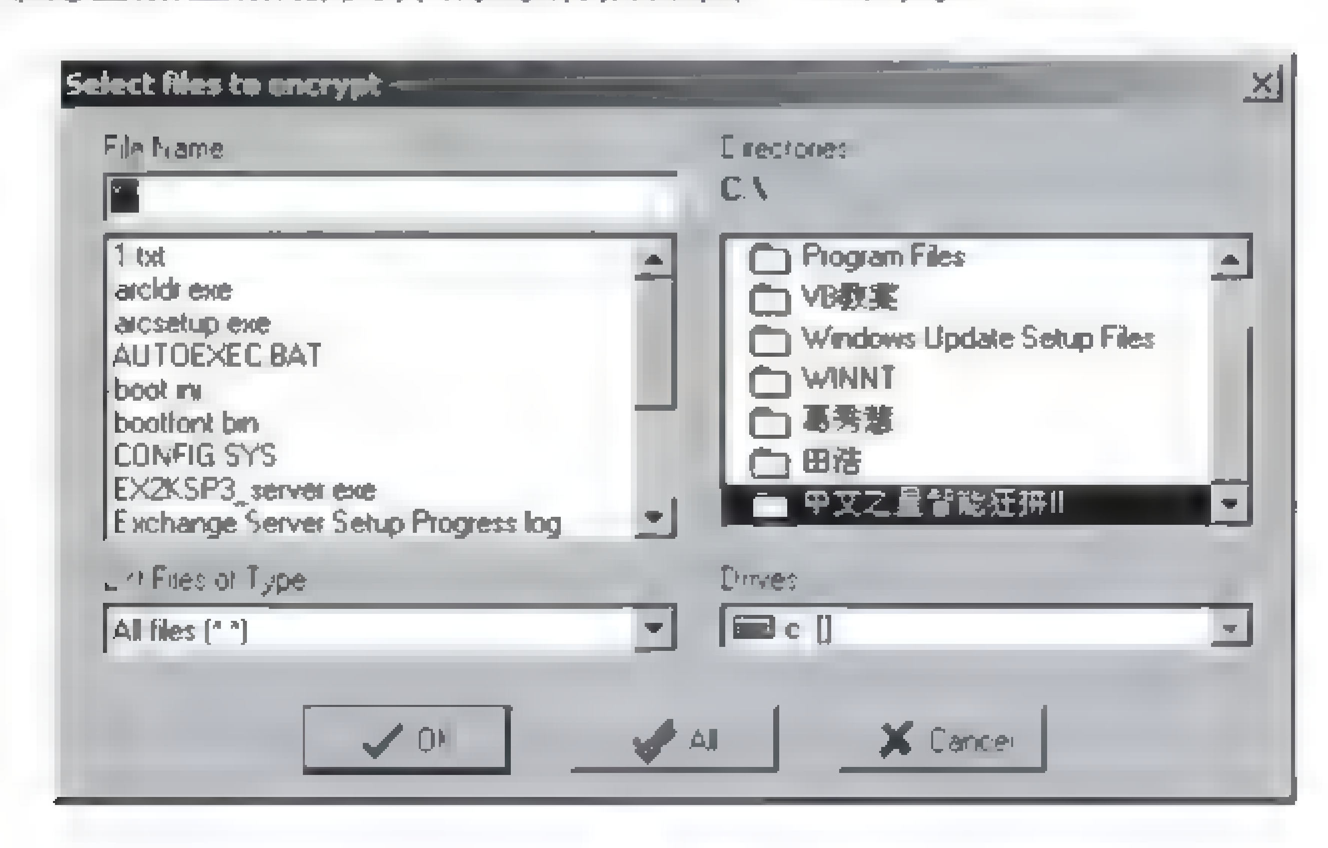


图　7-9

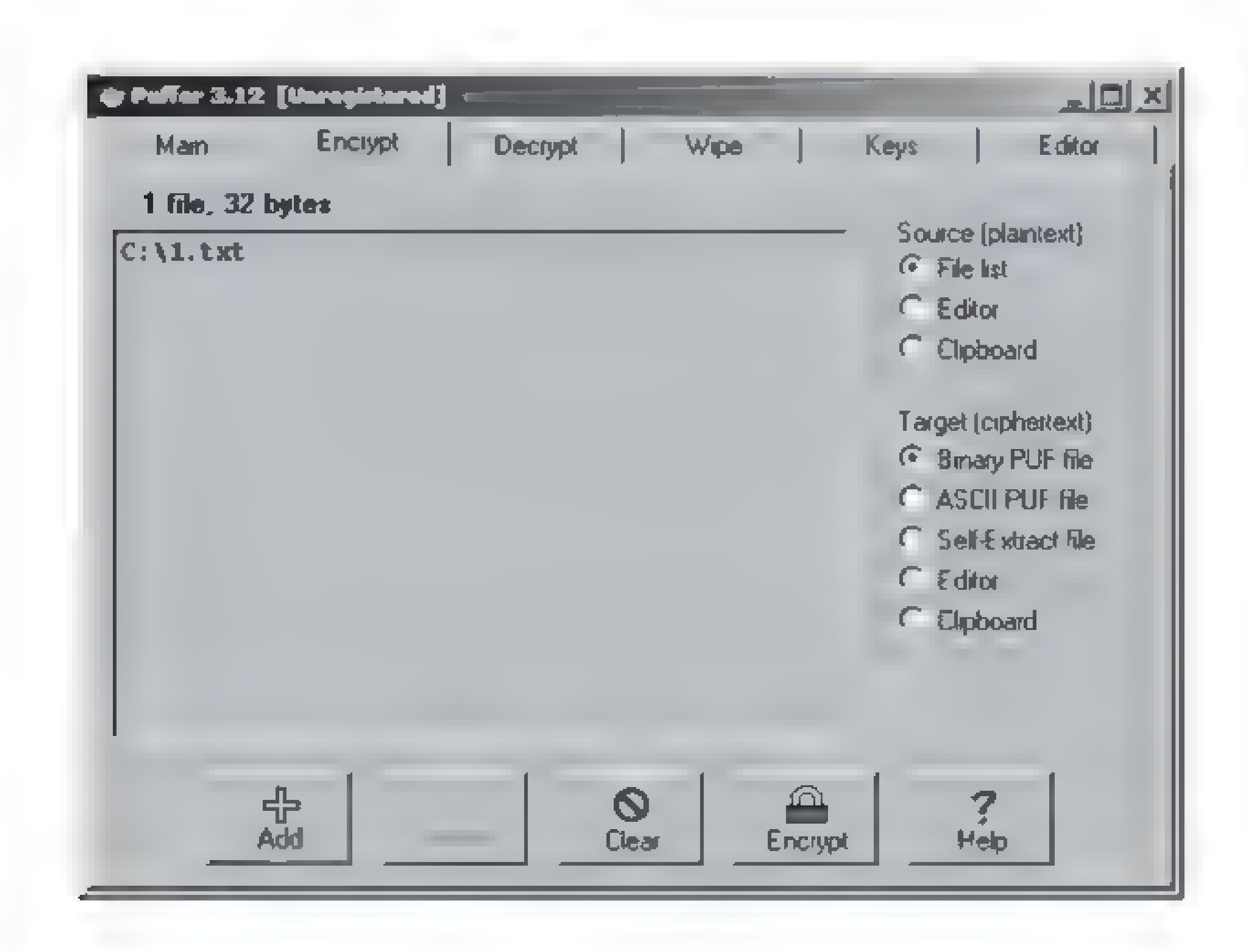


图　7-10

（3）单击 Encrypt 按钮，打开如图 7-11 所示的对话框，在 Encrypt with 选项区域中的单选按钮有如下两项。

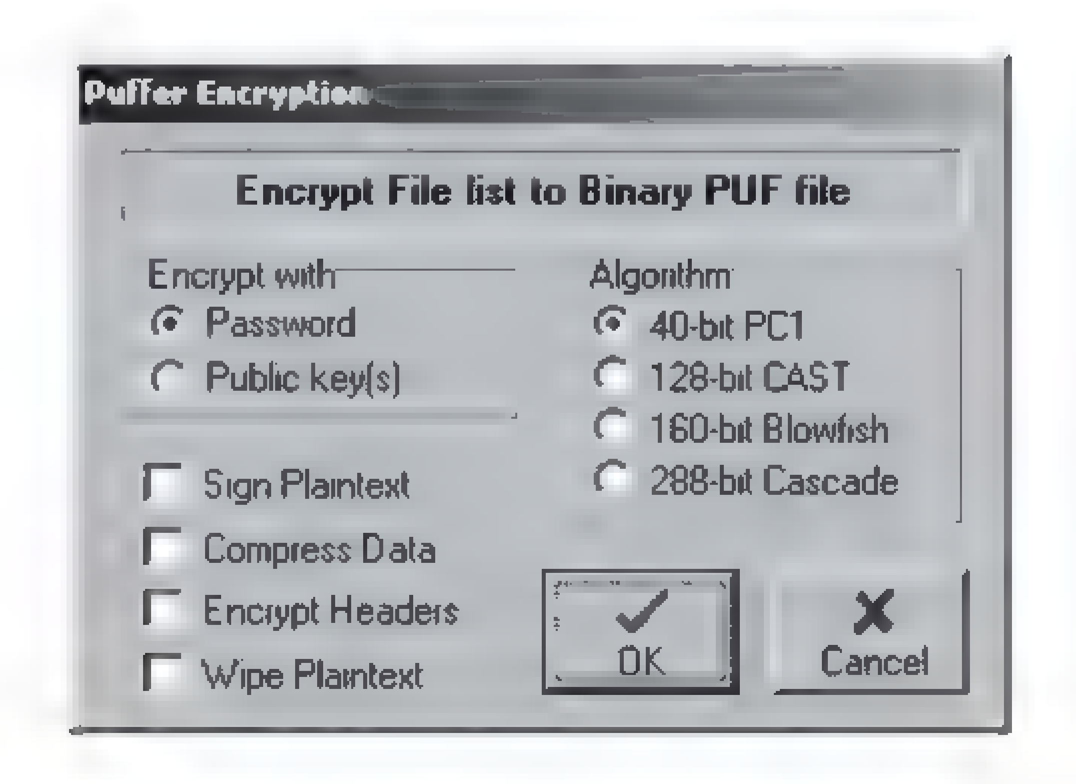


图 7-11

- Password：密码。
- Public key（s）：公钥加密。

在 Algonthm（算法）中可以选择 40 位、128 位、160 位和 288 位中之一的加密算法。

- Sign Plaintext：为文件进行数字签名。
- Compress Data：压缩文件数据。
- Encrypt Headers：加密信头。
- Wipe Plaintext：加密文件后删除原文件。

（4）选择完成后，单击 OK 按钮（这里采用默认值）将打开图 7-12 所示的对话框，输入保存的文件名。



图 7-12

（5）输入文件名和路径后，单击“保存”按钮将打开图 7-13 所示的 Encryption Password（加密密码）对话框，输入密码和确认密码，然后单击 OK 按钮。Password options 的三个选项说明如下。

- Case sensitive：区分大小写。

- 8 character minimum：最小为 8 个字符。
- Echo asterisks：是否用*来代替输入的字母出现。

2）用 Password 解密

当收件人收到加密文件后，利用 Puffer 来解密。具体操作为：

打开 Puffer，单击 Decrypt 菜单项，操作步骤和加密类似，只要知道加密密码即可。

2．Public key（公钥算法）

1）创建公钥

（1）创建自己的公钥，打开 Puffer，选择 Keys 选项卡，如图 7-14 所示。

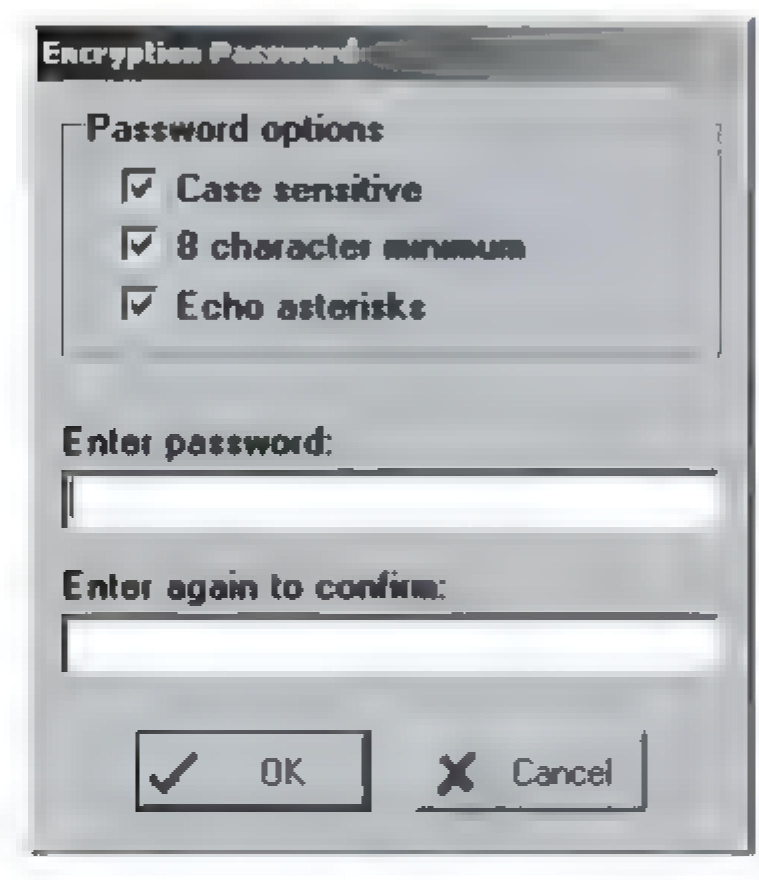


图　7-13

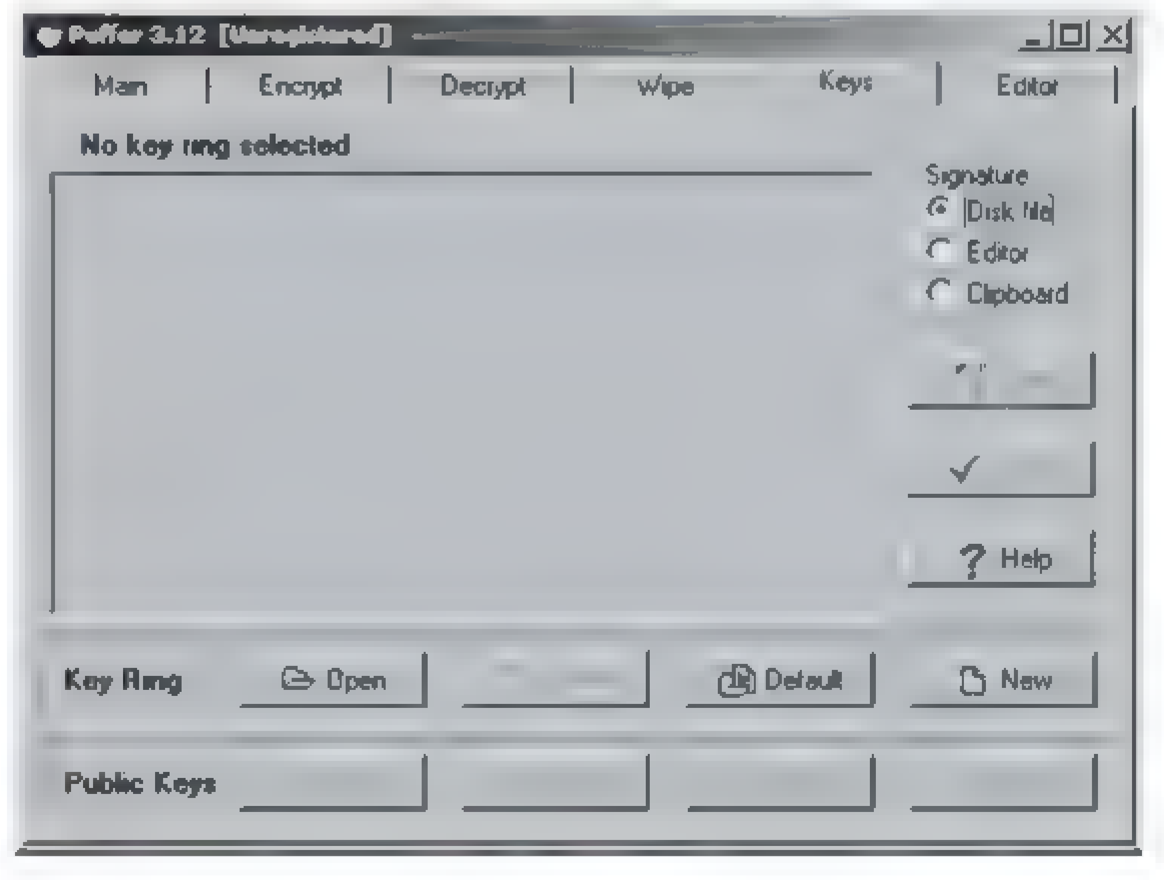


图　7-14

（2）单击 New 按钮，建立公钥的文件名，扩展名为.ppk，如图 7-15 所示。



图　7-15

（3）输入文件名后，单击“保存”按钮，打开图 7-16 所示的对话框。

（4）单击 Create 按钮，将出现图 7-17 所示的对话框，在 Your name 文本框输入姓名，在 E-mail 文本框中输入 E-mail 地址。在 Password 文本框中设置一个口令（不少于 10 位），这个口令要牢牢记住，日后才能打开用 Public key 法加密的文件。在 Confirm 文本框中再次输入口令。在 Key Expiration Date 中设定该 Public key 的有效使用期。

（5）单击 Create 按钮，打开图 7-18a 所示对话框，按 20 个键，生成 Key string，这是公钥字符串，要记住，以后公钥用于网上传输。



图 7-16

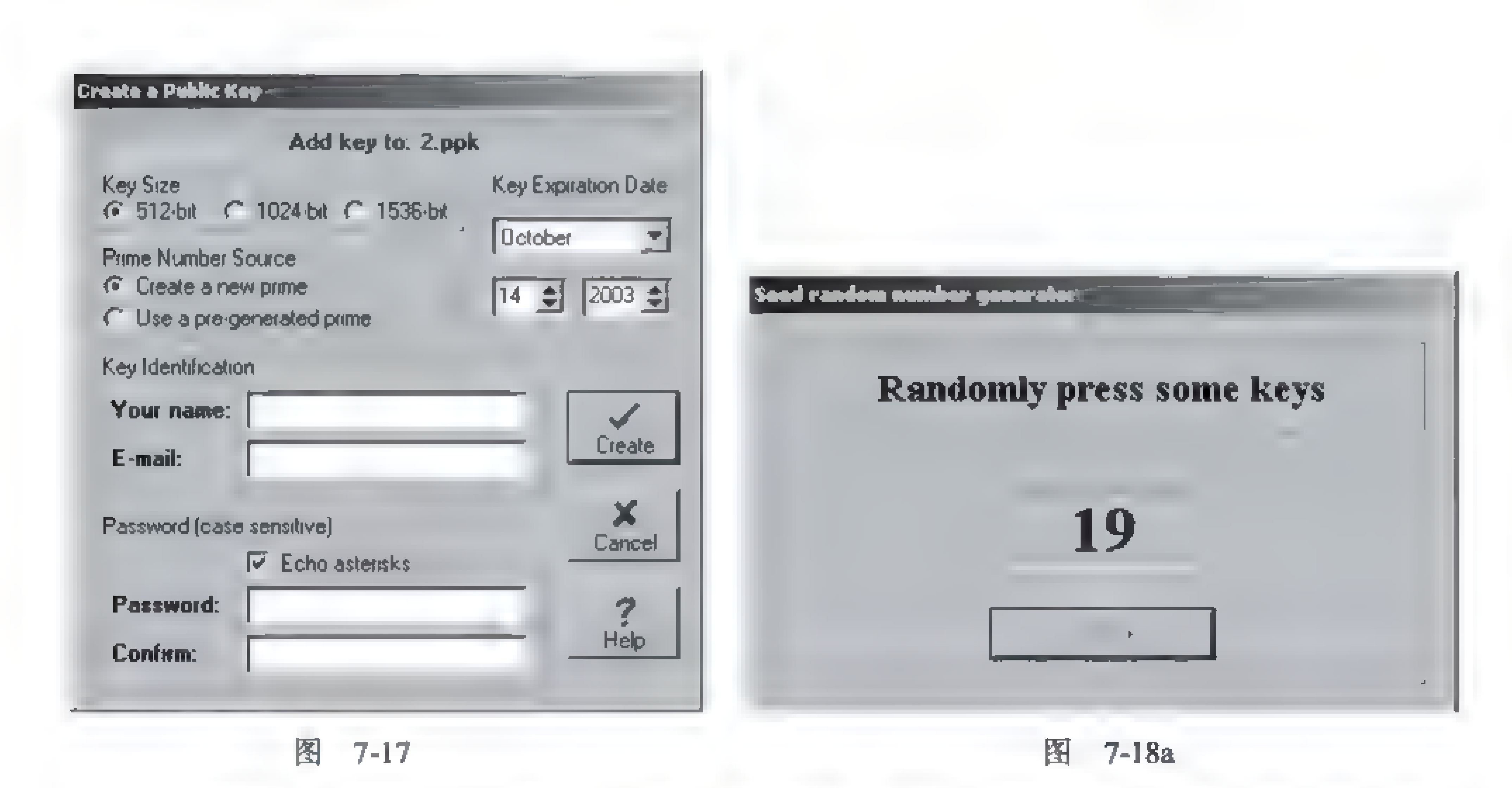


图 7-17　　　　图 7-18a

（6）单击 OK 按钮，打开图 7-18b 所示的对话框，记住自己的公钥，然后单击 OK 按钮。

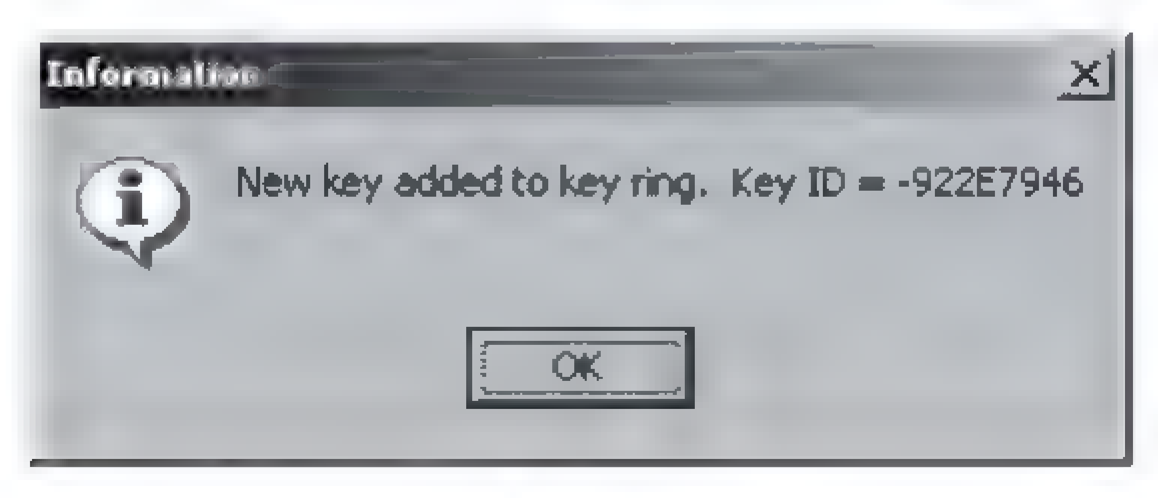


图 7-18b

2）加密

（1）加密操作的前几步同 Password，在图 7-19 的 Encryptwith 中选择 Public key。

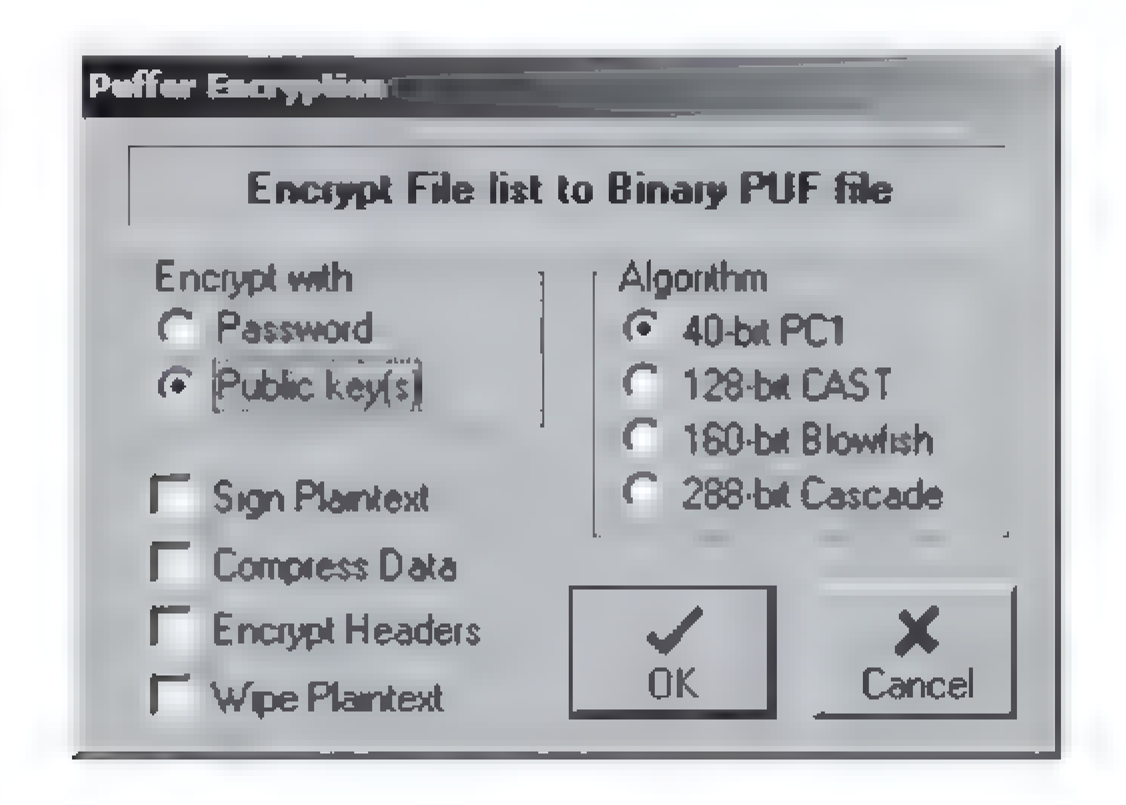


图　7-19

（2）单击 OK 按钮，确定生成文件名，如图 7-20 所示，单击“保存”按钮保存。

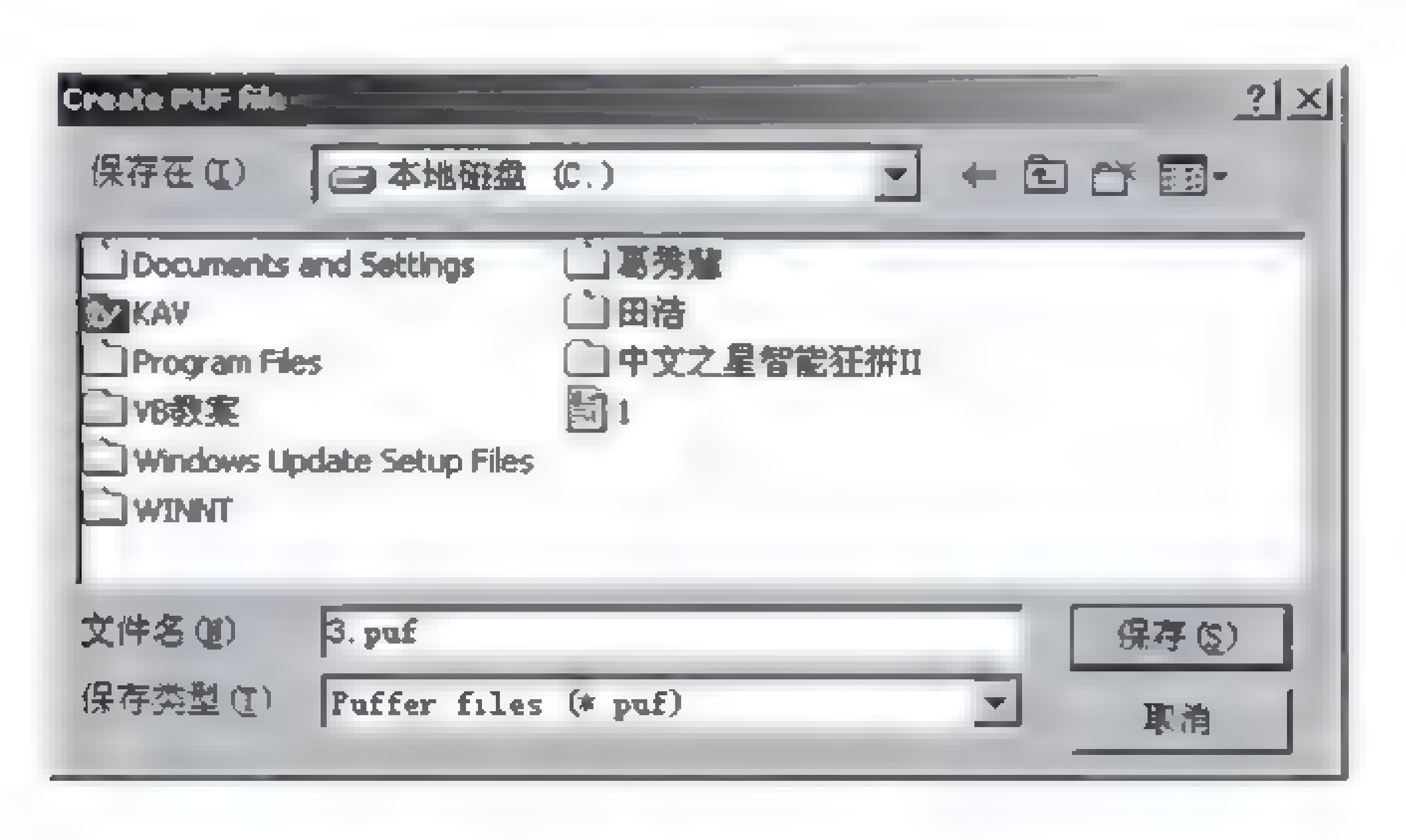


图　7-20

（3）单击 Key Ring，选中需要使用的公钥，然后单击 Add 按钮，如图 7-21 所示。

（4）可以看到公用密钥已经被加入到 Recipients 列表中，然后单击 OK 按钮，如图 7-22 所示，然后在出现的 Information 信息框中单击 OK 按钮，如图 7-23 所示。

3）用 PublicKey 解密

解密步骤的前几步与口令法一样，选择要解密的文件，再单击 Decrypt（解密）按钮，Puffer 会自动查出它是用公钥加密的，并弹出公钥解密对话框，如图 7-24 所示，在 Select your public key 下拉列表框中选择公共钥匙，在 Password 文本框正确输入密码，最后单击 Decrypt 按钮，在弹出的图 7-25 所示的 Information 信息框中单击 OK 按钮，就完成了解密。



图 7-21

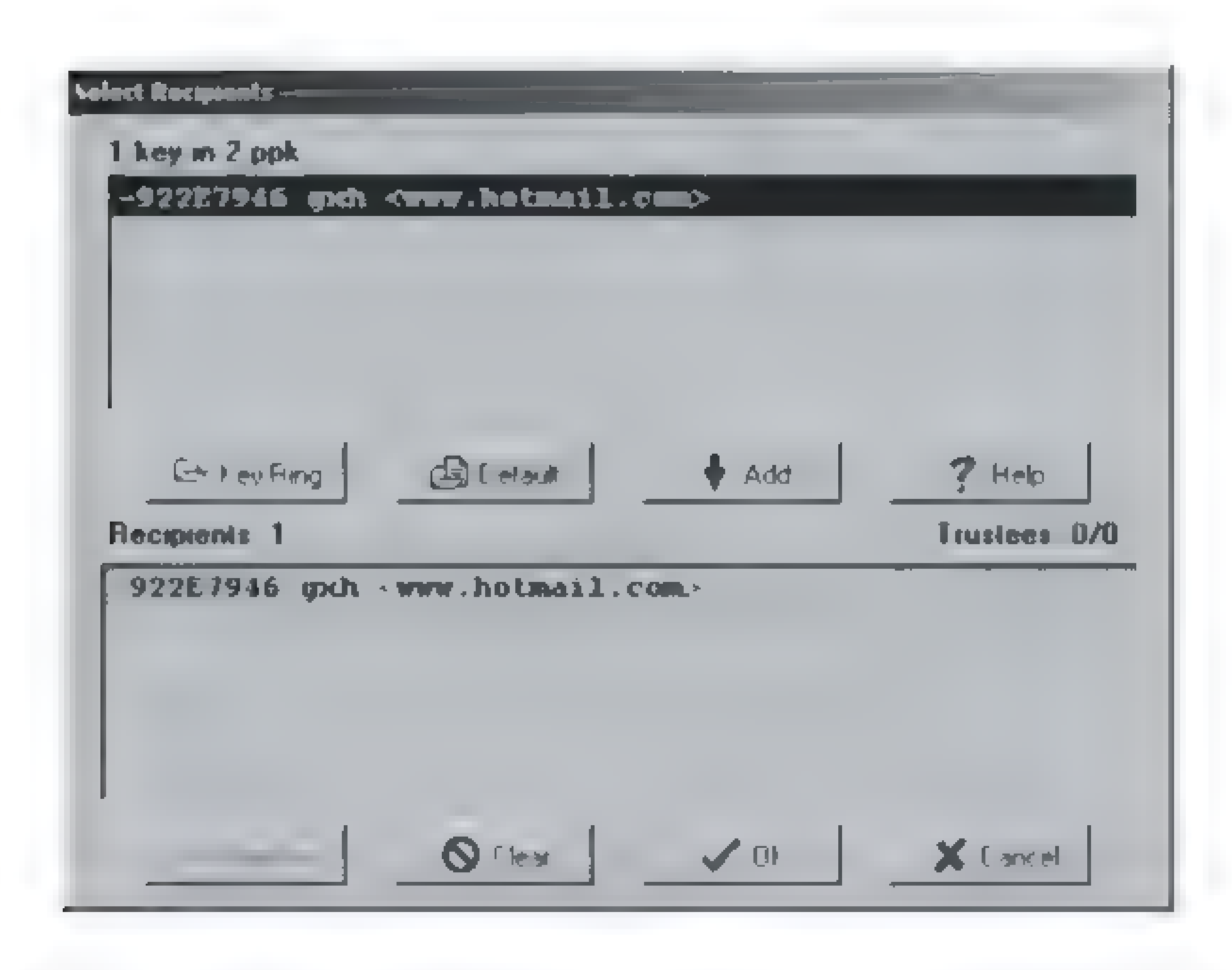


图 7-22



图 7-23

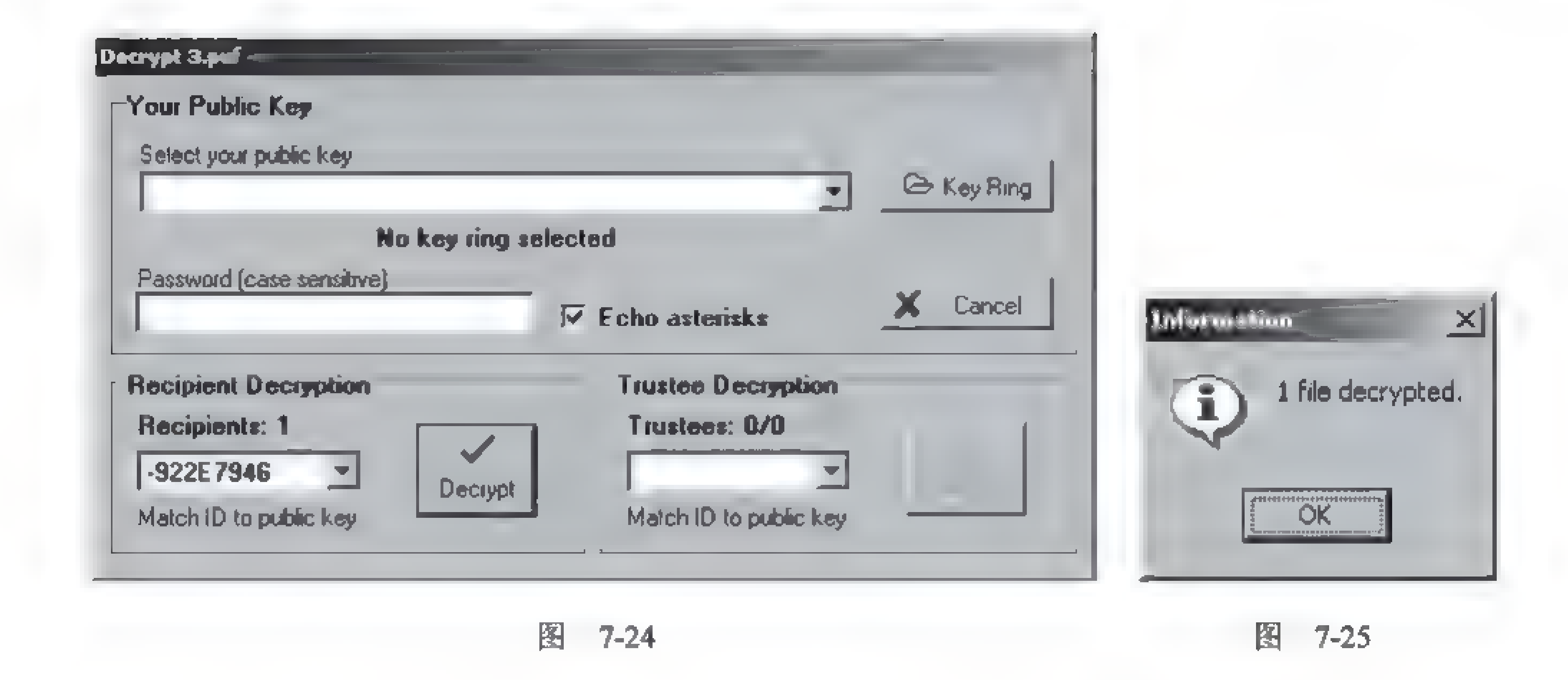


图 7-24　　　　图 7-25

7.5 Exchange 邮件服务器的安全配置与管理

本节将配置一个 Exchange 2000 邮件服务器，使之成为一个安全的邮件服务器。作为一个安全的邮件服务器，最好是专机专用。Exchange 是通过 MMC 管理控制台来提供管理的，选择“开始”→“程序”→Microsoft Exchange→System Manage 选项，即可打开管理控制台，如图 7-26 所示。

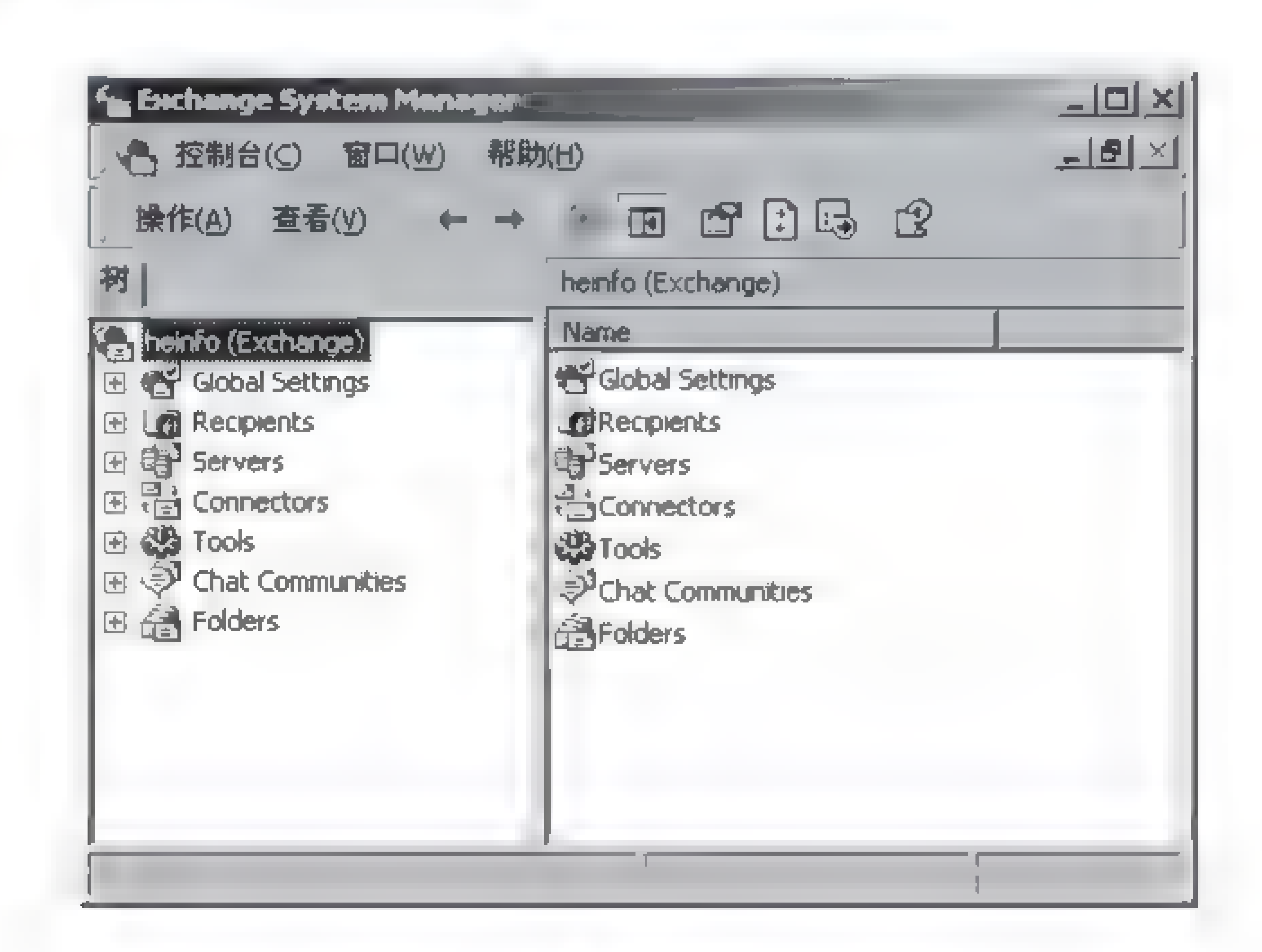


图 7-26

在这个控制台中主要完成以下几方面的管理。

- Global Settings：它包含三个方面的内容，如图 7-27 所示。

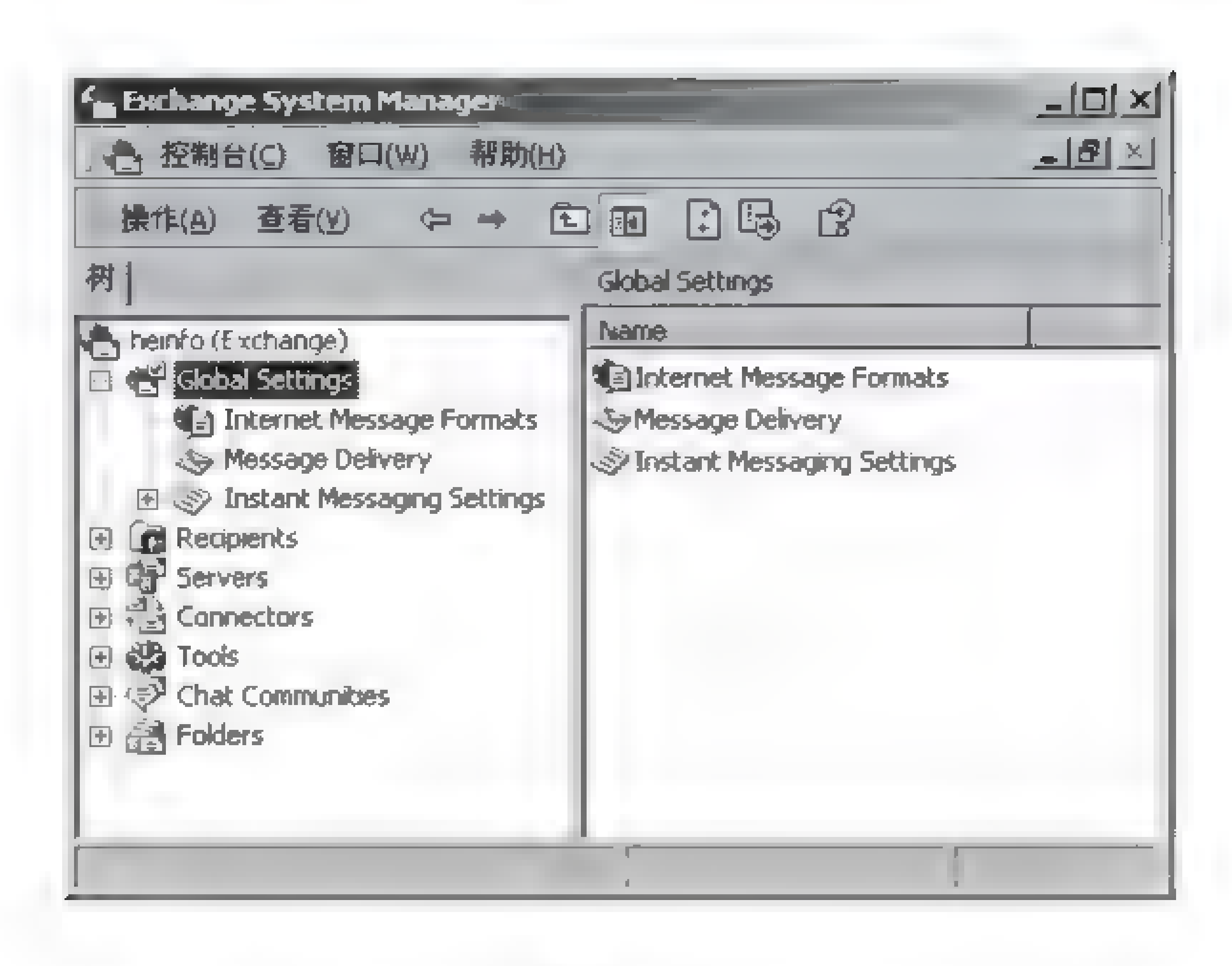


图 7-27

Instant Messaging Settings 定义了远程信息的 SMTP 信息传递格式，可以通过其属性对其进行相应的格式定义。

Message Delivery 完成邮件服务的相应设置；

Internet Message Formats 主要设置在 Internet 上的信息的传输格式。

- Recipients：收件人的管理，如图 7-28 所示，其中包含了组收件人、所有的地址列表和公共文件夹等收件人。在这里完成对接收的定义及地址管理。

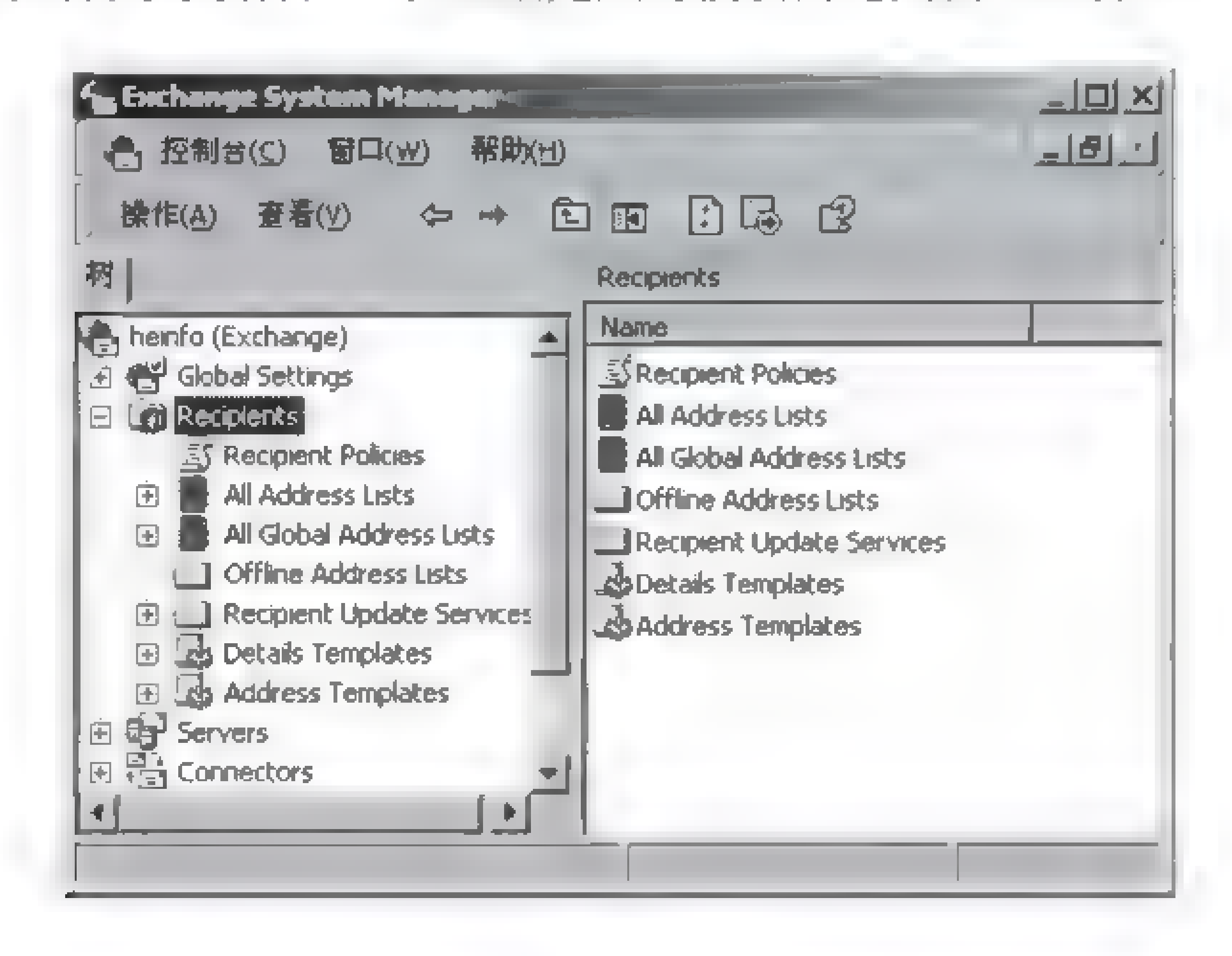


图 7-28

- Servers：主要是对服务器的一些管理工作，在此例中是对 DNS 及相关协议，以及邮箱的管理，如图 7-29 所示。
- Tools：这是帮助管理邮件服务的工具，可以配置带邮件服务的网点的收件人、邮件跟踪服务及邮件监控工具，如图 7-30 所示。

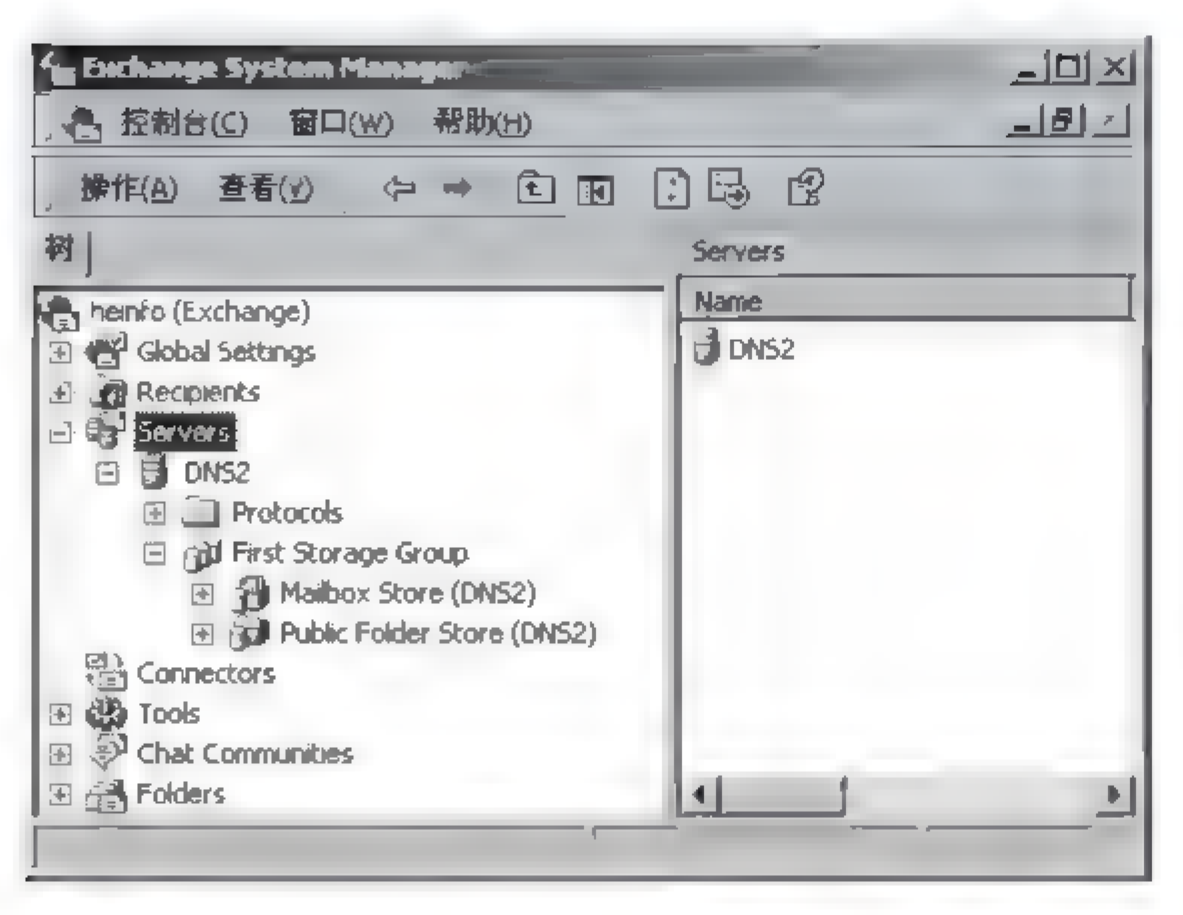


图　7-29

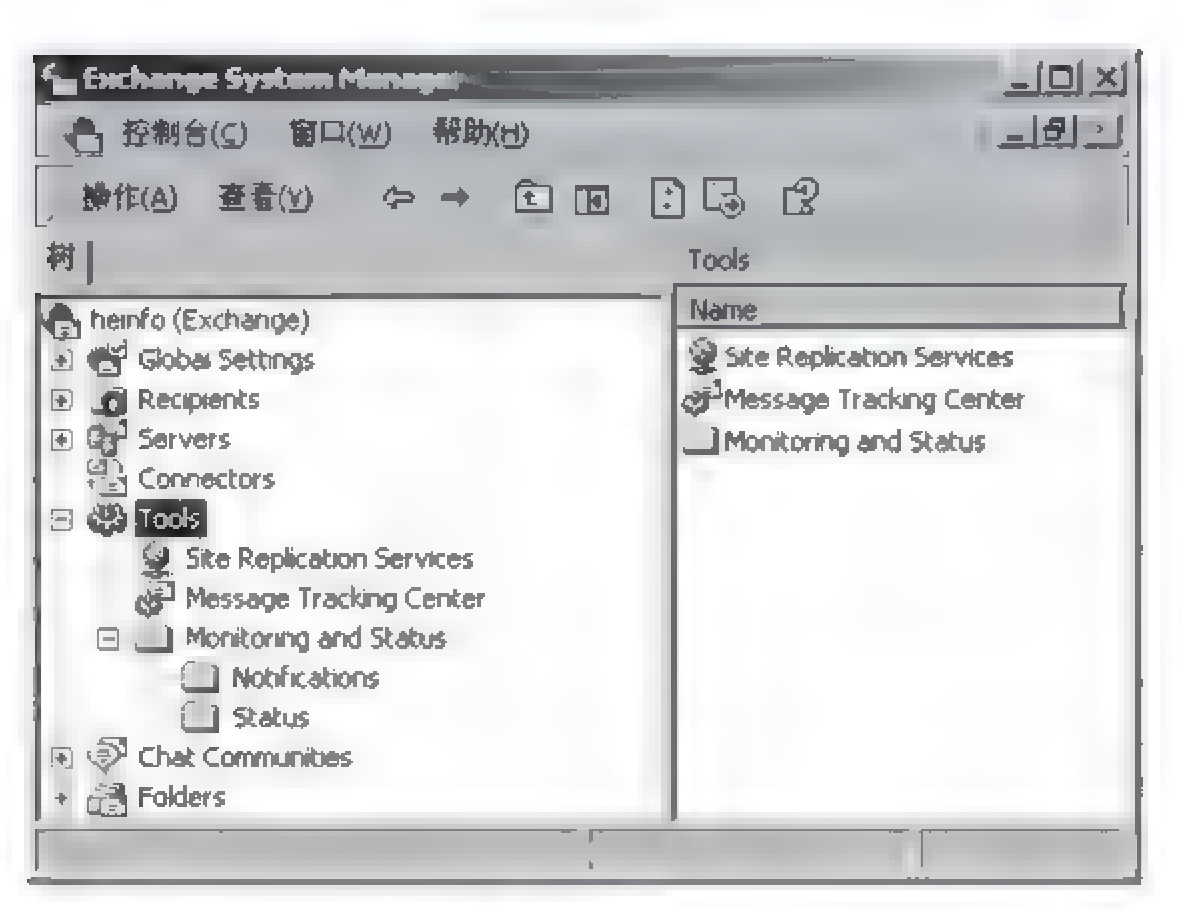


图　7-30

- Folders：管理公共文件夹是网络管理组的，如图 7-31 所示。

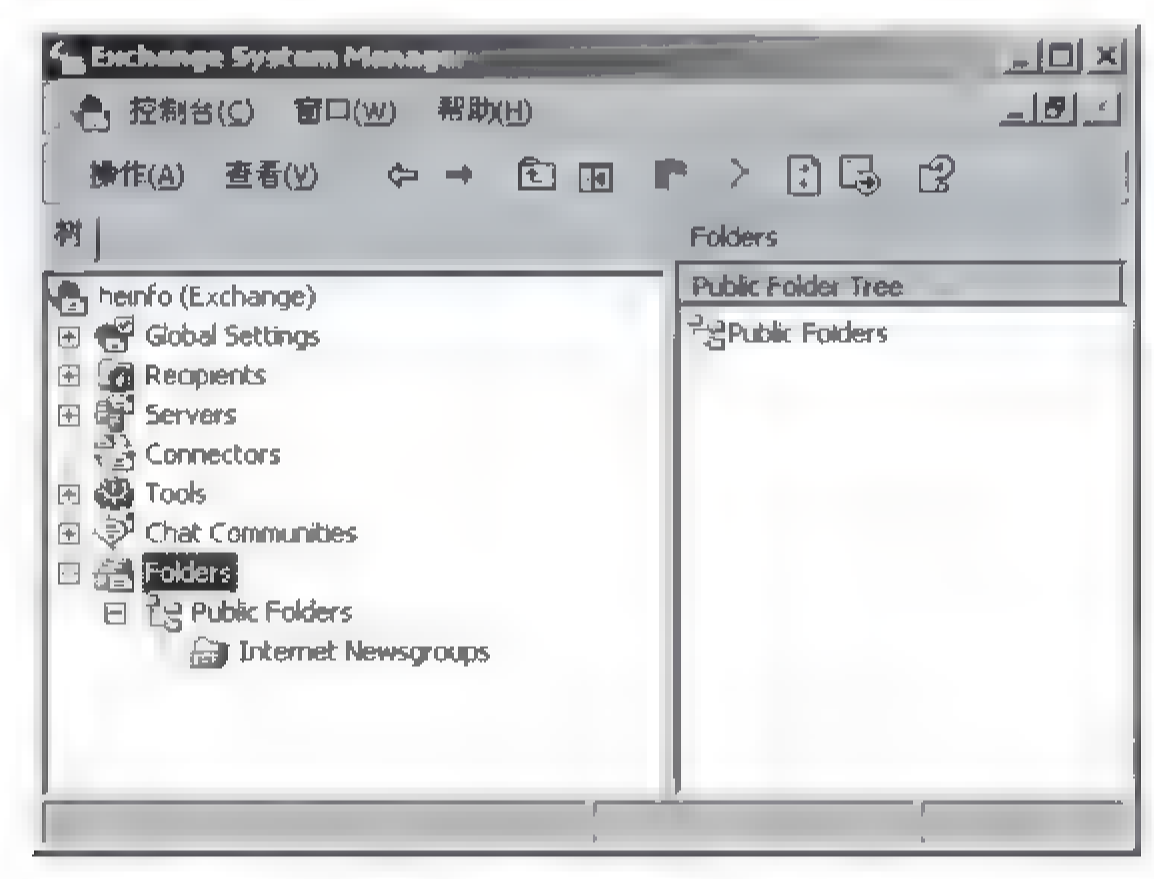


图　7-31

7.5.1 收件人的创建与配置

在 Exchange 2000 的用户可以有两类邮箱的配置：邮箱允许和邮件允许。在每创建一个新用户时就会自动地为该用户创建一个邮箱，可以对这个邮箱进行配置。在 Exchange 中有 4 类收件人：用户、联系人、组和公用文件夹，这里将以用户为例讲述收件人的创建与配置。

1. 收件人的创建

创建的具体步骤如下。

（1）选择"开始"→"程序"→Microsoft Exchange→Active Directory Users and Computers→Users 选项，在 Users 容器处右击，在弹出的快捷菜单中选择"新建"→"用户"命令，打开图 7-32 所示的"新建对象-用户"对话框。

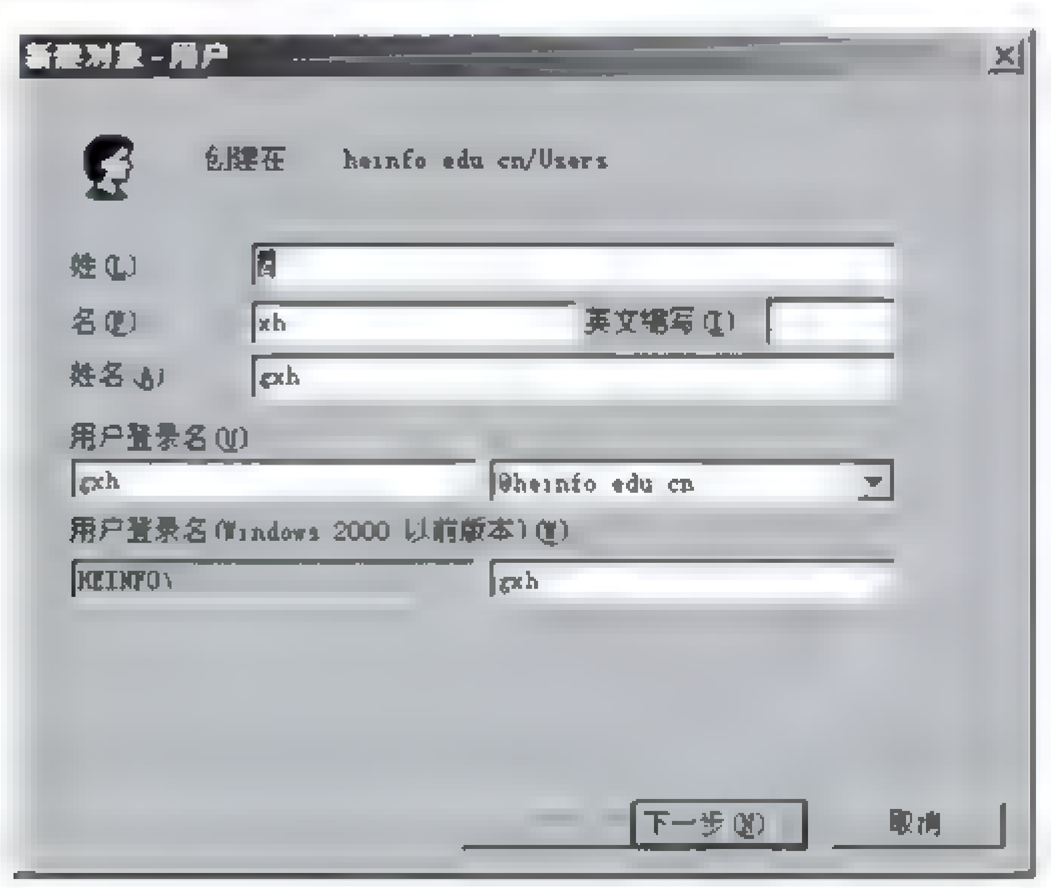


图 7-32

（2）在对话框中完成相应文本框的填写后，单击"下一步"按钮，打开图 7-33 所示的确认密码对话框。

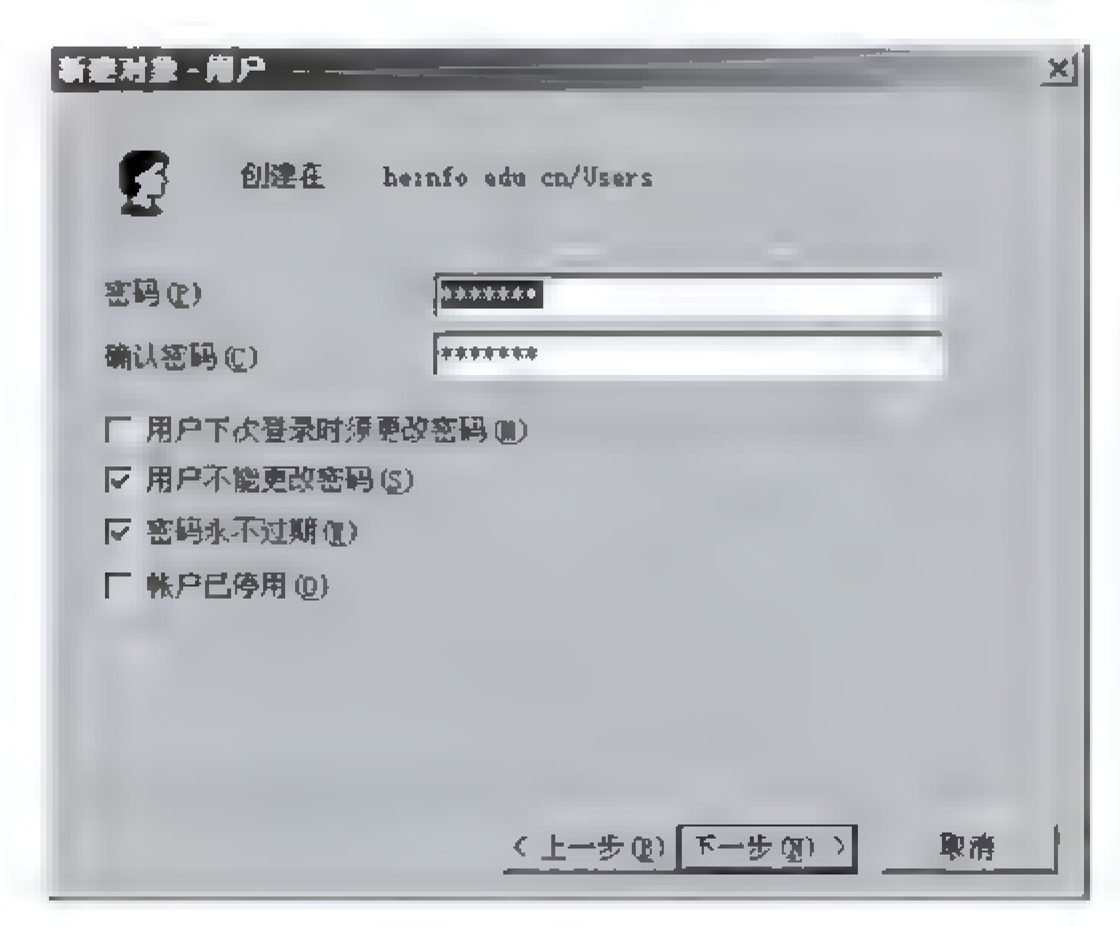


图 7-33

（3）单击"下一步"按钮，打开图 7-34 所示的创建邮件服务的邮箱对话框，在其中选择自己的别名、服务器及邮箱的存储位置，也可采用默认值。



图　7-34

（4）单击"下一步"按钮，打开图 7-35 所示的完成新建用户对话框，单击"完成"按钮，这样在创建用户的同时为用户创建了一个邮箱。

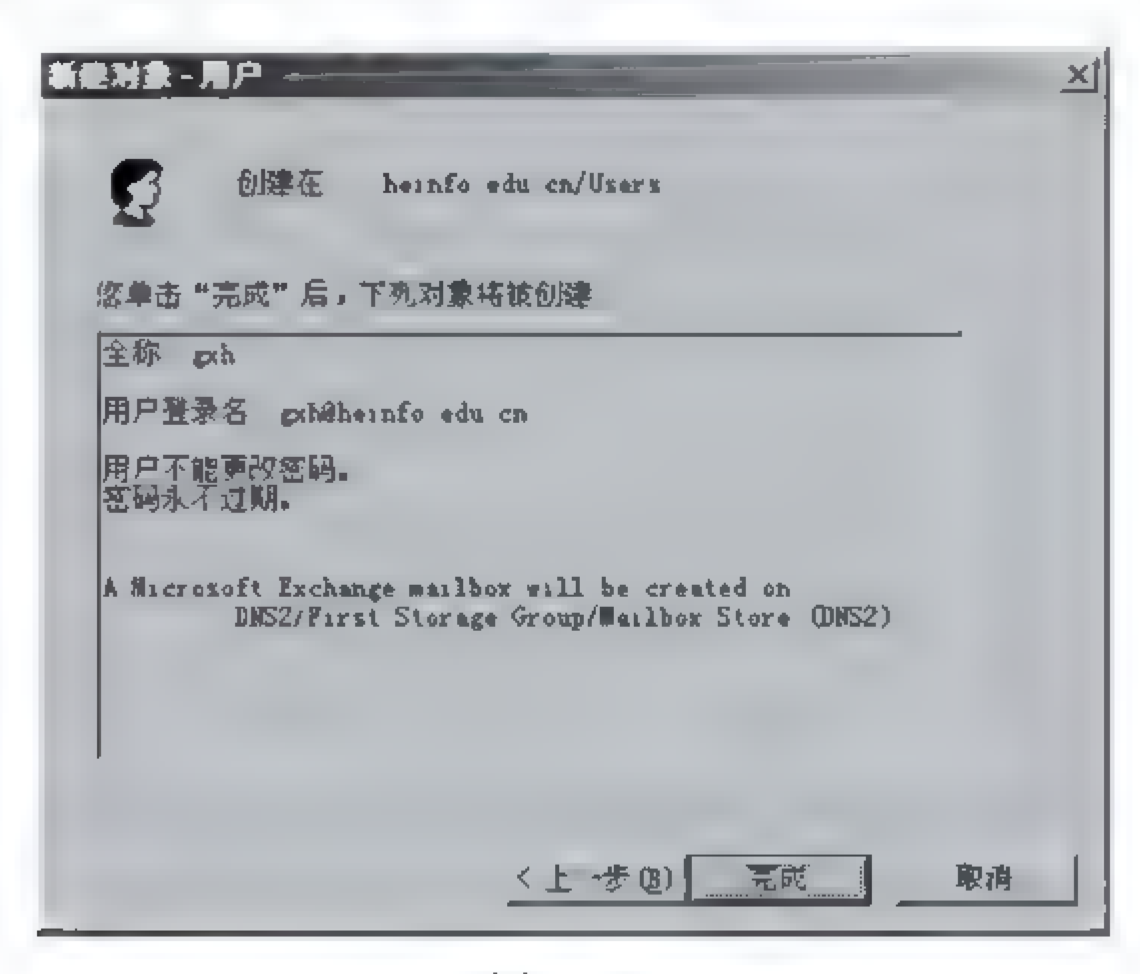


图　7-35

2．收件人的配置

对上述已经建成的 gxh 用户的邮箱进行配置，右击用户名，在弹出的快捷菜单中选择"属性"命令，打开图 7-36 所示的属性对话框。

在图 7-36 中可以对 Delivery Restrictions、Delivery Options 和 Storage Limits 三项进行修改，单击 Delivery Restrictions 按钮，将出现图 7-37 所示的 Delivery Restrictions 对话框，在此可以设置发送信息字节的大小、接收信息的大小及对信息的接收方式，可以在对话框

中指定，也可以采用默认值。

图 7-36

图 7-37

当单击 Delivery Options 按钮后，在出现的图 7-38 所示的 Delivery Options 对话框中可以对目的地址、收件人限制等进行修改。

当单击了 Storage Limits 按钮后，在出现的图 7-39 所示的 Storage Limits 对话框中，可对存储及删除的方式进行限制。

图　7-38

图　7-39

另外，在属性中还可以对 E-mail Addresses、Exchange Features 和 Exchange Advanced 选项卡等进行修改。

对于 E-mail Addresses 选项卡，可以为邮箱从不同的邮件通信地址中接收消息进行配置，一个邮箱对同一类型可以有多个地址，如图 7-40 所示。

图 7-40

在图 7-41 所示 Exchange Advanced 选项卡中，可以通过 Custom Attributes、Protocol Settings、ILS Settings 和 MailBox Rights 按钮进行对应选项的设定。

图 7-41

4 种设置对应的选项卡分别如图 7-42、图 7-43、图 7-44 和图 7-45 所示，根据实际所需进行相应配置。

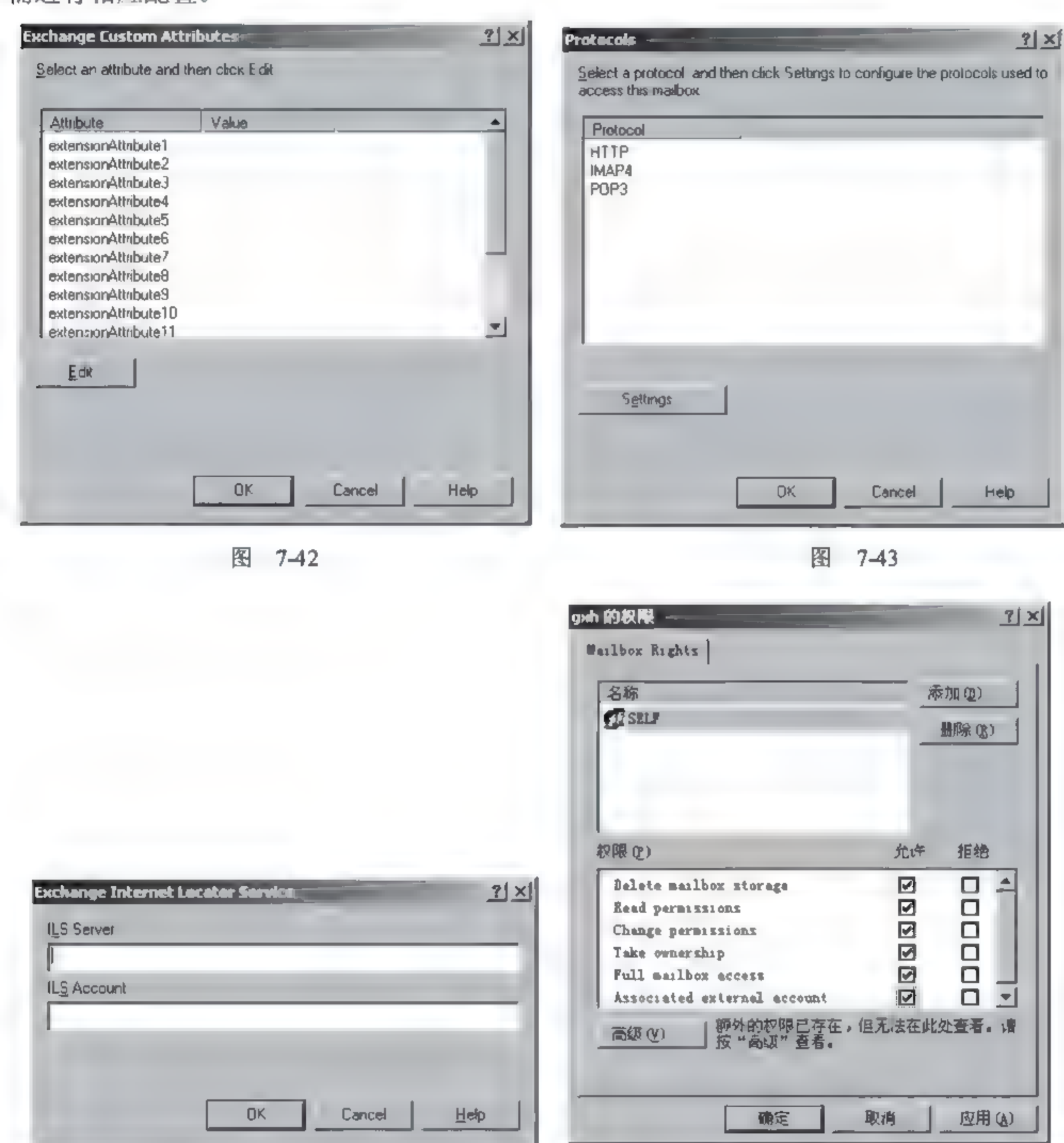


图 7-42　　图 7-43

图 7-44　　图 7-45

3. 过滤收件人

选择“开始”→“程序”→Microsoft Exchange→Active Directory Users and Computers 选项后，再选择“查看”→“筛选器选项”选项，出现图 7-46 所示的“筛选器选项”对话框，在此对话框中可以进行筛选，如果选择了创建“自定义筛选器”单选按钮，则出现图 7-47 所示的“查找自定义搜索”对话框，在对话框中可以自己定义筛选器。

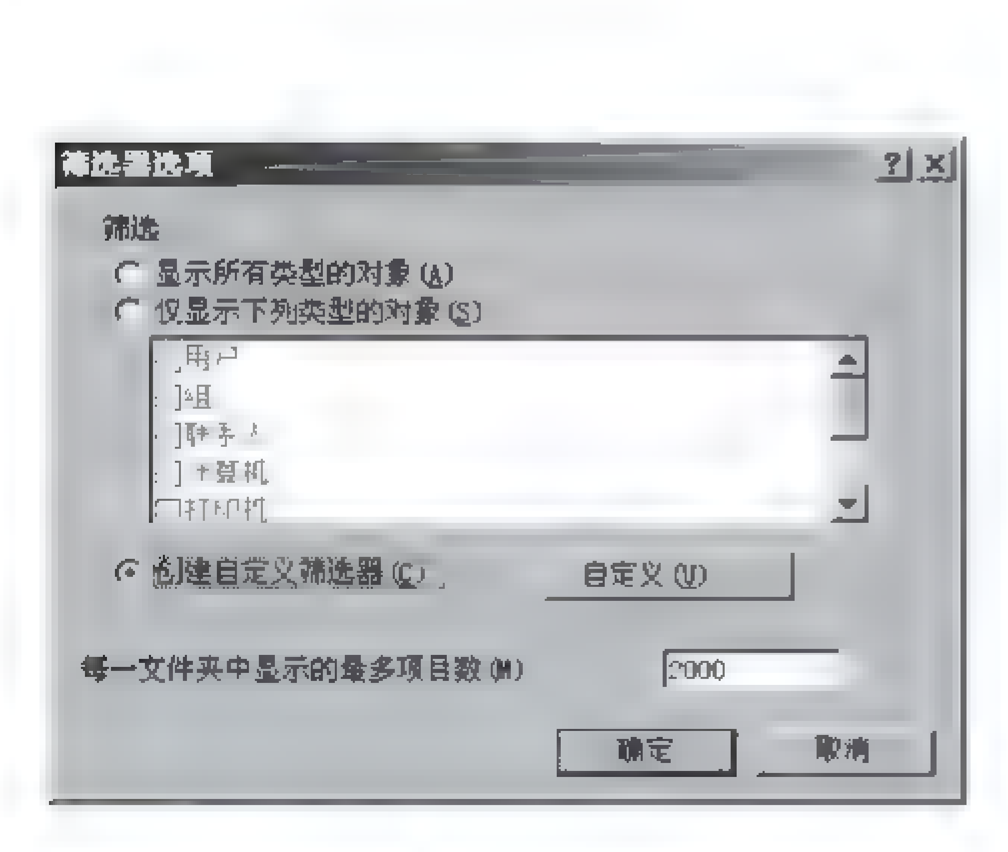


图 7-46

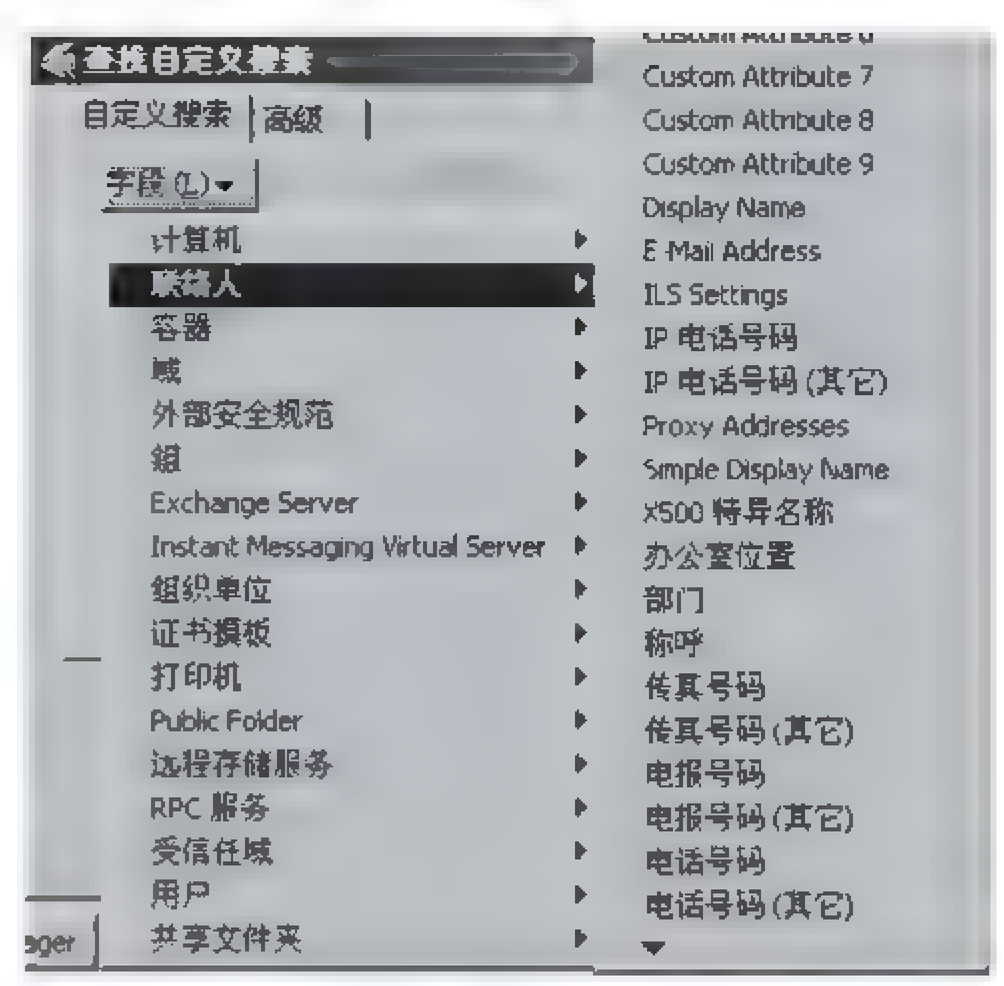


图 7-47

7.5.2 Exchange Server 的监控

邮件服务器的监控是网络运行成功的关键，所以经常使用一些工具来完成这些操作。这里介绍两种常用的工具事件浏览器和系统监视器。

1. 事件浏览器

选择“开始”→“程序”→“管理工具”→“事件查看器”选项，将出现图 7-48 所示的“事件查看器”对话框。

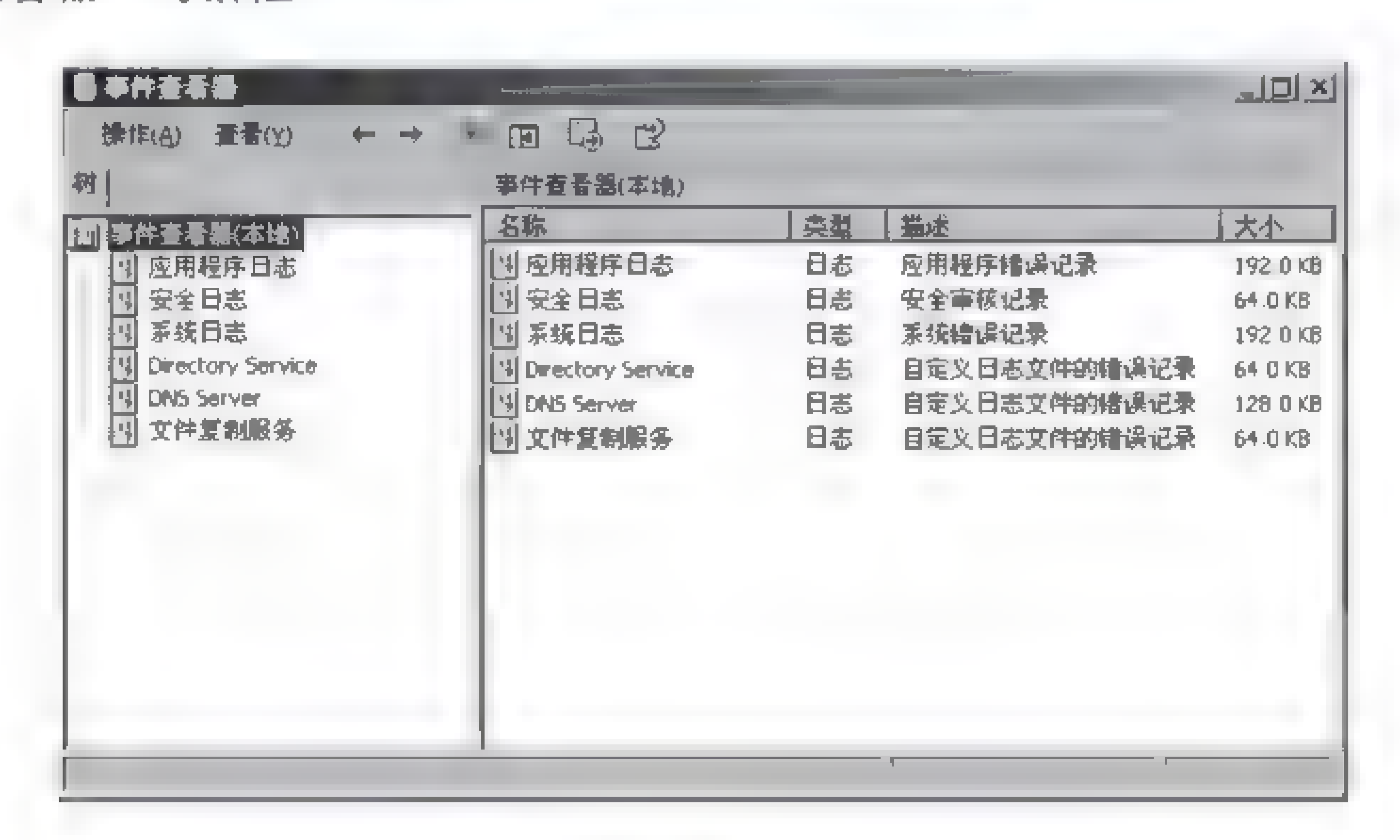


图 7-48

事件查看器中共有 6 类事件的日志：应用程序日志、安全日志、系统日志、Directory Service（目录服务）、DNS Server（DNS 服务）和文件复制服务。

在日志中有 5 种事件类型：成功审核、错误、信息、警告和审计失败，如图 7-49 所示。



图 7-49

- 成功审计：指一次安全审计访问尝试。
- 错误：出现邮件服务没有正确启动等重大问题。
- 信息：描述成功操作的重要事件。
- 审计失败：指一次安全审计访问尝试失败。
- 警告：当前系统无害的事件，但将来可能会引发问题。

2. Exchange 监视器

选择“开始”→“程序”→Microsoft Exchange→System Manager→Tools→Monitoring and Status 选项，打开图 7-50 所示 Exchange System Manager 对话框，这就是 Exchange 监视器。

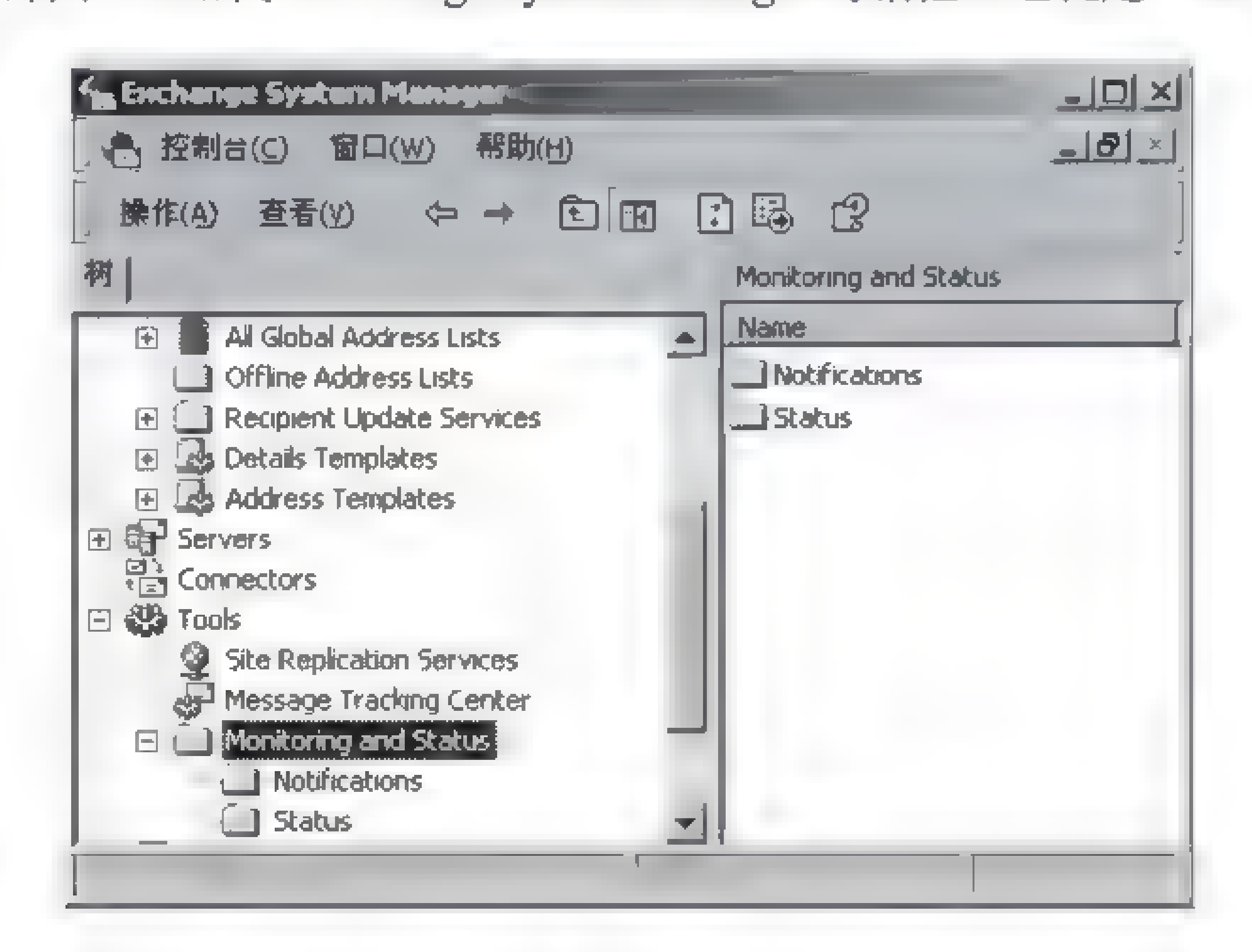


图 7-50

使用 Exchange 监视器，网络管理员可以通过邮件服务器中的邮件进行管理和监视，它能捕获发送出的和发送到本邮件服务器的全部数据的帧。

第8章 计算机病毒

21 世纪计算机网络技术飞速发展，人们正在享受着由此带来的种种便利。然而由于种种原因，也有人在编制各种各样的计算机病毒，给人们的正常的生活和工作带来了无数的麻烦，甚至造成了不可挽回的损失。因此，了解和掌握有关计算机病毒的知识显得越来越重要了。

在本章中，将详细探讨一下计算机病毒的基本概念、种类、破坏力以及常见病毒的分析、预防和常用杀毒软件的使用。

8.1 计算机病毒概述

谈“毒”色变并不为过，经常使用计算机的人大部分都受到过病毒的“恩泽”，小到数据莫名其妙的丢失，大到整个计算机系统瘫痪，其破坏能力使人震惊。在本小节中将对计算机病毒作全面概括性的叙述。使读者有全面而翔实的理解。

8.1.1 计算机病毒的定义

在《中华人民共和国计算机信息系统安全保护条例》中明确指出：计算机病毒是指编制或者在计算机程序中插入的破坏计算机功能或者毁坏数据，影响计算机使用，并能自我复制的一组计算机指令或者程序代码。

计算机病毒的特点是，它是人为的特制程序，具有自我复制能力，很强的感染性，一定的潜伏性，特定的触发性和很大的破坏性。

计算机系统的信息需要存储、读取、复制和传送，这就为计算机病毒的传播提供了途径，计算病毒就会伴随着计算机系统的正常的存储、读取、复制、传送而繁殖、感染、破坏。

当今的计算机病毒往往利用计算机操作系统自身的弱点进行攻击和传播，提高计算机系统的安全性是防病毒的一个非常重要方面。病毒与反病毒将作为一种技术对抗长期存在，两种技术都将随计算机技术的发展而得到长期的发展。

8.1.2　病毒的产生

1）最早的计算机病毒

计算机的先驱者冯·诺伊曼（John Von Neumann）这个名字对每一个学过计算机的人都不陌生。他不仅提出了计算机的结构，而且早在 1949 年第一商用计算机出现之前好几年，在他的一篇论文《复杂自动装置的理论及组织的进行》里，已经勾勒出病毒程序的蓝图。不过在当时，绝大部分的计算机专家都无法想象会有这种能自我繁殖的程序。

计算机病毒的概念来自一场游戏。1977 年美国著名的 AT&T 贝尔实验室中，三个年轻的程序员——道格拉斯·麦耀莱（Douglas McIlroy），维特·维索斯基（Victor Vysottsky）及罗伯特·莫里斯（Robert T. Morris），当时年纪都只有二十多岁。在工作之余，玩一种游戏：彼此撰写出能够吃掉别人程序的程序来互相作战。这个叫做"磁芯大战（core war）"的游戏，进一步将计算机病毒"感染性"的概念体现出来（Robert T. Morris 就是后来写了一个蠕虫病毒，把 Internet 搞得天翻地覆的那个 Robert T. Morris Jr. 的爸爸，当时的 Morris 刚好是负责 Arpanet 网路安全）。

1982 年匹兹堡的一名高中生编写了一个恶作剧程序，通过软盘在苹果机的操作系统中传播，并显示一首歪诗。这个名为 ElkCloner 的病毒被看作是计算机领域的第一个病毒。

而 Morris 是第一个通过网络传播的计算机病毒。在 1988 年 11 月 2 日由麻省理工学院（MIT）的学生 Robert Tappan Morris 撰写。该病毒总共仅 99 行程序代码，施放到当时网络上数小时，就有数以千计的 UNIX 服务器受到感染。但此软件原始用意并非用来瘫痪电脑，而是希望写作出可以自我复制的软件，但程式的循环没有处理好，使得服务器不断执行、复制 Morris，最后死机。

2）当今流行计算机病毒的产生的原因

可以肯定计算机病毒完全是人为的特制程序，所有的计算机病毒都是由人为故意编写的，甚至多数病毒可以找到开发者的个人信息。

病毒的开发者编写病毒的主要情况和目的有以下几种。

- 计算机专业人士为表现自己和证明自己的能力而编写的特殊的程序。如 CIH 病毒等。
- 对上司或社会不满，为了满足自己的报复心理，如"熊猫烧香"病毒。
- 处于对自主开发的软件的产权保护而预留的陷阱。
- 用于特殊目的。为了达到军事目的或某组织的特殊目的，而编写的破坏性程序。

8.1.3　计算机病毒的特征

计算机病毒的种类很多，但通过病毒代码的分析比较可知，它们的结构是相似的，都包括三个部分：引导部分、传染部分和表现部分。

引导部分是将病毒加载到内存，作好传染的准备；传染部分将病毒代码复制到目标；表现部分根据特定的条件触发病毒。概括讲，计算机病毒具有的特征为如下几点。

- 传染性：计算机病毒有很强的再生机制，病毒程序一旦加到运行的程序体上，就能感染其他程序，并且迅速扩散到整个计算机系统，当与网络进行数据交换时，也将

病毒在网上传播。

- 寄生性：病毒依附在其他程序体内，当该程序运行时病毒就进行自我复制。
- 潜伏性：计算机病毒入侵系统后，一般不立即发作，而具有一定的潜伏期。当时机成熟才发作。
- 隐蔽性：计算机病毒的传播都没有外部表现，它是隐蔽在正常程序中，另外，病毒都是具有很高编程技巧的短小精悍的程序，如不经过代码分析，很难识别正常程序和病毒程序之间的区别，不到病毒发作，很难发现。
- 破坏性：病毒的破坏能力是不一样的，有的占用系统资源，有的造成计算机硬件损坏，有的修改或删除文件及数据等。

8.1.4 病毒的分类

目前对计算机分类的方法有很多，下面介绍常见的分类方法。

1．按病毒寄生方式分类

根据病毒寄生方式分类，病毒可分为网络型病毒、可执行文件型病毒、引导型病毒及复合型病毒。

- 网络型病毒：通过计算机网络传播、感染网络中的可执行文件。
- 可执行文件型病毒：主要是感染可执行文件。被感染的可执行文件在执行的同时，病毒被加载并向其他正常可执行文件传染。
- 引导型病毒：主要感染软盘、硬盘的引导扇区或主引导扇区，在用户对软盘硬盘进行读写操作时进行感染。
- 复合型病毒：不仅传染可执行文件而且还传染硬盘引导区，被这种病毒传染的系统用格式化命令都不能消除此类病毒。

2．按病毒的破坏后果分类

根据病毒破坏的能力可划分为以下几种。

- 良性病毒：干扰用户工作，但对计算机系统无损。
- 恶性病毒：这类病毒删除程序、破坏数据、清除系统内存区和操作系统中重要的信息。

3．按病毒的发作条件分类

按病毒的发作条件分为以下几种。

- 定时发作型：具有查询系统时间功能，当系统时间等于设置时间时，病毒发作。
- 定数发作型：具有计数器，能对传染文件个数等进行统计，当达到数值时，病毒发作。
- 随机发作型：没有规律，随机发作。

4．按连接方式分类

- 源码型病毒：主要攻击高级语言编写的源程序。

- 入侵型病毒：主要攻击特定的程序，用自身替代部分模块或堆栈区。
- 操作系统型病毒：主要攻击操作系统，用自身替代操作系统功能。
- 外壳型病毒：主要是附在正常程序的开头或结尾。

8.1.5 计算机病毒的发展

从计算机诞生到现在，计算机病毒也和计算机发展一样经历了几个阶段，具体如下。

1. DOS 引导阶段

计算机发展初期，计算机病毒主要是指引导型病毒，引导型病毒利用修改启动扇区而获得对计算机的控制权。

2. DOS 可执行阶段

在引导型阶段之后，就是可执行病毒和复合病毒阶段，这种可执行病毒在系统执行文件取得控制权，然后感染系统的可执行文件（如.exe、.bat 和.com 类可执行文件）。

3. 变体阶段

这个时期有许多种类型病毒出现，主要为生成同文件名的伴随型文件病毒和多种变体的病毒。

4. Windows 阶段

随着视窗在个人 PC 上的广泛使用，感染它的病毒日渐其多，主要是利用其操作系统的保护模式及 API 调用接口及相关的系统漏洞。

5. 网络阶段

随着网络的普及，网络已经成为病毒的最佳传播途径，大部分的病毒借助于网络这一现代化的工具而迅速在全世界传播，而邮件的各类病毒更是不胜枚举。所以现在的病毒都是属于网络阶段的病毒。

8.1.6 计算机病毒的破坏现象

知道计算机病毒出现的特征，才能更好地查杀这些病毒，下面介绍常见病毒发作时，计算机产生的相关现象。

- 引导时死机。
- 引导失败。
- 开机运行几秒后突然黑屏。
- 外部设备无法找到。
- 计算机出现异样声音。
- 计算机处理速度明显变慢。

- 系统文件字节变化或系统日期发生改变。
- 驱动程序被修改。
- 计算机经常死机或重新启动。
- 应用程序不能进行一些必要操作。
- 系统内出现大量文件垃圾。
- 文件的大小和日期改变。
- 系统的启动速度慢。
- 键盘、打印、显示有异常现象。
- 文件突然丢失。
- 系统异常死机的次数增加。

8.2 常见的几种病毒及其查杀方法

下面介绍几种常见病毒的查杀方法。

8.2.1 CIH 病毒

每月的 26 日，是 CIH 病毒的发作日，它是首例破坏计算机系统硬件的病毒，是一种破坏性很强的恶性病毒，该病毒的核心是用 VXD（虚拟设备驱动程序）技术编制的，具有很强的实用性和隐蔽性。另外，此种病毒的变体在不断增加，所以对 CIH 病毒有了深刻了解，才能更好地使自己的系统免受此类病毒的攻击。

CIH 病毒感染 Windows 95/98/ME 等操作系统的可执行文件，能够驻留在计算机内存中，并据此继续感染其他可执行文件。CIH 的危险之处在于，一旦被激活，它可以覆盖主机硬盘上的数据并导致硬盘数据丢失。并且它还具备对主机 BIOS 的攻击能力，使计算机无法引导。目前大多数主板仍然使用 EEPROM 作为 BIOS 芯片，此种芯片自身没有任何安全的保护措施。对付 CIH 病毒可以采用以下两种方法：一是制作 BIOS 的硬备份，即使用 ROM 编程器备份自己的 BIOS；二是软备份，因为 CIH 并没有改变 ROM 的内容，而只是在 EEPRAMK 增加了 1 个字节，因此可以使用编辑可执行文件，恢复自己的 BIOS。具体步骤如下：

首先，制作一张 DOS 系统盘，将该损坏主板相同型号的 BIOS 和刷新文件复制到系统盘上。然后，在系统盘的根目录下自动批处理文件中，加入@ echo off, a:\AWARD.EXEX.BIN X，这里的 X 代表损坏主板的 BIOS 文件名。这样将使用系统盘重新引导，一般都能恢复计算机的 BIOS。

此外，CIH 病毒还向硬盘不断地写入数据，造成硬盘原有数据的丢失。一般恢复有以下两种方法。

1. 用急救盘

对于被破坏的硬盘可以用一些相关杀毒软件的紧急恢复来完成。下面以 KILL 为例，讲述如何恢复硬盘数据：

- 恢复第一个逻辑盘。第一个通常是 C 盘，将 KILL 放入软驱，进入急救盘菜单，选择比较修复引导区，然后修复（如果已经制作过急救盘，按以上步骤进行操作；如果没有制作过急救盘，那么需要找一台和此台具有同样配置的机子，制作一张急救盘）。
- 可以用 NDD、Debug 类工具恢复其他逻辑盘。

另外，CIH 还可以破坏硬盘，对于遭到破坏的主板的修复可以做如下处理：

- 如果是品牌机，与厂家联系。
- 如果是兼容机，则找到一个相同型号的主机，下载主板厂家提供的升级文件。

2. CIH 的手工清除

现有的杀毒软件均可以查杀 CIH 病毒，除了使用专业杀毒软件之外，还可以用手工的方式清除，方法如下：

- 在染毒文件中找特征字串。

查找 5ECC568BF0，把 CC 改为 90；查找 5ECCFB33DB，把 CC 改为 90。

- 在染毒文件中找。

查找 CD205300010083C420，改为 909090909090909090；查找 CD2068004000，改为 909090909090。

8.2.2　木马病毒

木马，又名特洛伊马（Trojan），源于古希腊的特洛伊马神话，传说希腊人围攻特洛伊城，一直未果，后来用了木马计，把士兵藏于木马之中，其他军队假装撤离而将木马丢于特洛伊城下，敌人将木马拖入城内，木马内的士兵从木马中爬出与城外的军队里应外合攻下了特洛伊城。

木马的危害性在于对计算机系统强大的控制和破坏能力，窃取密码、控制系统操作、进行文件操作等，一个功能强大的木马一旦被植入机器，攻击者就可以像操作自己的机器一样控制该机器，甚至可以远程监控对该机的所有操作。

1. 木马的入侵

木马由客户端和服务器端两个执行程序组成，客户端是用于攻击者远程控制植入木马的机器，服务器端程序即是木马程序。攻击者要通过木马攻击计算机系统，首先需要把木马程序植入到攻击目标的计算机内，一般通过下载和匿名邮件等方式侵入目标计算机，木马执行文件很小，一般以 K 字节为单位，所以在下载或收发一些邮件时往往下载了木马而不会察觉。另外一种常见的入侵方式是通过脚本植入（如 Script、ActiveX 及 Asp、Cgi 等）。由于应用软件不可避免地存在漏洞，这些漏洞就是攻击者的目标，如 Script 脚本漏洞对浏览者硬盘进行格式化的 Html 页面等。

2. 木马的激活

为了控制入侵目标计算机，木马要激活自己而自我运行。激活的方式一般有以下几种。

1）在 Win.ini 中启动

在 Win.ini 文件的[Windows]字段中的启动命令 Load=和 Run=一般都为空，如果其后面有了可执行文件 run=c:windowsX.exe，X 代表文件名，那这些可执行文件就可能是木马程序。

2）修改文件关联

修改文件关联是木马常用手段，关联大家都知道，当双击一类的文件名，它都会打开相应的应用程序，如双击.txt 文件，将打开记事本文件。那么木马是如何修改关联的？它是通过修改 HKEY_CLASSES_ROOTcomfileshellopencommand 下的键值，如将 c:windowsnotepa.exe%1 改为 c:windowssystemsysexplr.exe%1，当双击一个文本文件，这时就是激活木马程序，所以经常检查 HKEY_CLASSES_ROOT 中的文件类型可以发现并查杀木马病毒。

3）捆绑文件

有时，木马程序被查杀，但又会反复出现，这就必须考虑它已经和某类应用程序进行了捆绑，如和系统文件进行了捆绑，那么每次启动时都会加载木马程序。一般要注意以下几个文件。

System.ini：此文件在 Windows 的安装目录下，它的[boot]字段为 shell=explorer.exe，当 shell=explorer.文件名.exe 时，通常这个文件就是木马程序。

System.ini：此文件在 Windows 的安装目录下，它的[386enh]字段 driver=完整的文件名（包括盘符、路径、文件名），当不是你本身的驱动时，通常这个驱动就是木马程序。

System.ini：此文件在 Windows 的安装目录下，它的[mic][drivers][drivers32]字段都可以捆绑木马程序。

所以随时注意此系统文件，就可以有效地查杀木马病毒。

4）利用注册表

在注册表中所有以 Run 开头的键值，当与安装时的注册表异样时，一定要注意，这通常就是木马程序了。

5）利用自动批处理、系统配置文件和 WINstar.bat

一定要时常检查自己的自动批处理和系统配置文件，防患于未然。

3. 木马病毒的入侵检测

1）检测木马病毒前的准备

- Windows 设置：通过修改“我的电脑”→“工具”菜单→“文件夹选项”→“查看”选项卡—“隐藏受保护的操作系统文件”去掉前面选项钩的方法，显示隐含文件。如图 8-1 所示。
- 选中“显示所有文件和文件夹”复选项，如图 8-2 所示。
- 去掉“隐藏已知文件类型的扩展名”复选项前面的钩。

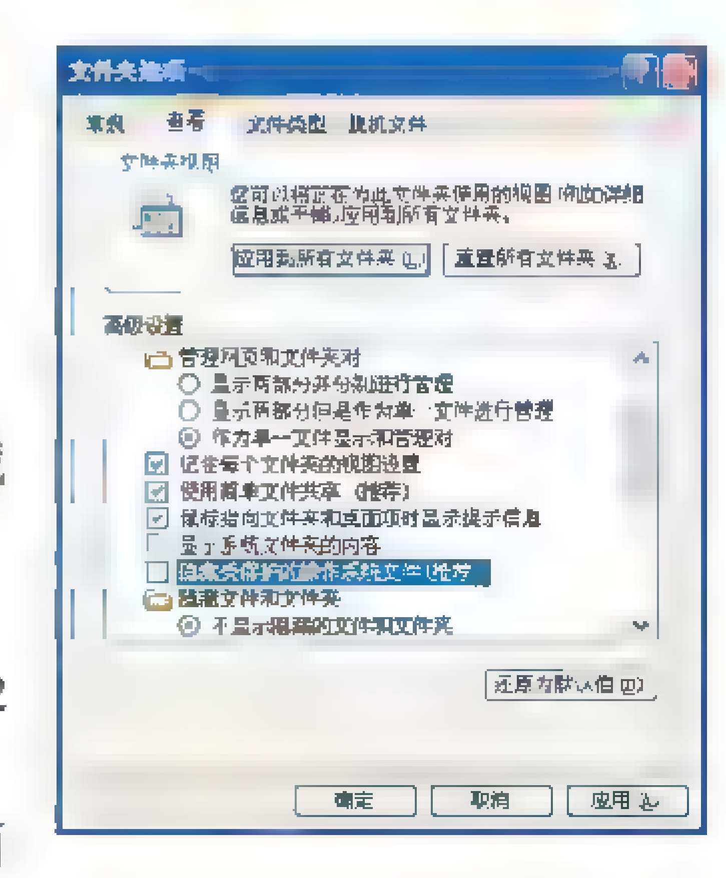


图 8-1

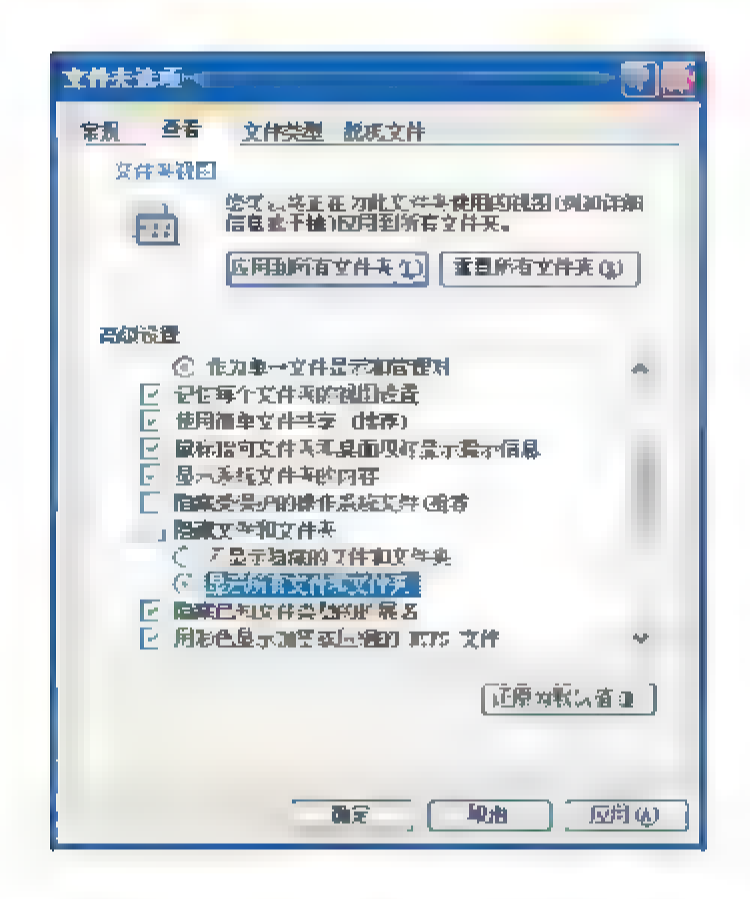


图　8-2

如果这时仍然不能看到隐藏文件，说明病毒已经修改了注册表。这种现象的解决办法是：

打开注册表编辑器，即在“开始”→“运行”中输入 regedit 后按回车键，在注册表中首先找到：HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Advanced\Folder\Hidden\NOHIDDEN，把右边框中有两个项目 CheckedValue 和 DefaultValue 的值全改为 2。

再找到：KEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Advanced\Folder\Hidden\SHOWALL，把 CheckedValue 的值改为 1，DefaultValue 的值改为 2，或者这两个值都改为 1，即可显示隐藏文件了。

但是，如果木马病毒仍然在运行，有可能上面的操作会被木马病毒还原，使得注册表修改失效。如果是这样，就要查看系统进程了。

2）手工检测木马病毒

对应木马病毒入侵和激活的特点，手工检测木马病毒必须具备三方面的知识：进程、注册表和服务。下面一一介绍。

首先介绍“进程”。简单地说，进程是程序在计算机上的一次执行活动。当一个程序在运行，系统就启动了一个进程。进程可以分为系统进程和用户进程。凡是用于完成操作系统的各种功能的进程就是系统进程，它们就是处于运行状态下的操作系统本身；用户进程就是所有由用户启动的进程。也就是说，进程是操作系统当前运行的执行程序。在系统当前运行的执行程序里包括：系统管理计算机本身和完成各种操作所必需的程序；用户开启、执行的额外程序，当然，也就可能包括用户不知道，而自动运行的非法程序——木马病毒。危害较大的可执行病毒同样以“进程”形式出现在系统内部（一些病毒可能并不直接显示在进程列表中，如“宏病毒”，但是可以通过第三方的工具，将隐藏的进程显示出来），因此及时查看并准确杀掉非法进程对于手工查杀各类木马和病毒起着关键性的作用。

举一个例子：svchost.exe 这个进程，大家应该都知道它的正确的路径应该是在 c:\windows\system32\svchost.exe。也就是说，如果发现有的 svchost.exe 进程其路径不是这个，那么就有 99.9%的把握确定它是木马病毒了。如果它的进程变成了 c:\windows\svchost.exe，那么只要先结束或终止该进程，再找到该进程路径所指的文件 c:\windows\

svchost.exe 将其删除，就可以达到清除该木马病毒的目的了。当然由于种种原因，可能从任务管理器中无法看到此进程的路径，或者结束不了进程，或者找不到病毒文件，这时就要借助第三方工具了——可以查看进程路径的软件（冰刃、DoIt 等）。既然在进程里存在，而按照进程路径却找不到文件，那么病毒肯定做了手脚，把自身设为隐藏文件，而且使系统不能显示隐藏文件，这里就用到了前面介绍的准备工作的知识了。找到经过隐藏的病毒文件后删除它就达到了清除病毒的目的。

其次介绍“注册表”。注册表中是一个数据库，包含 Windows 在运行期间不断引用的信息，例如，每个用户的配置文件、计算机上安装的应用程序以及每个应用程序可以创建的文档类型、文件夹和应用程序图标的属性表设置、系统上存在的硬件以及正在使用的端口。注册表非常复杂，在这里没有必要非常深入地讨论，但是要手工查杀病毒，至少应了解注册表中那些经常被木马和病毒利用的自启动项。可以通过:“开始”→“运行”→regedit 打开注册表编辑器。以下所列各项均是病毒经常利用的自启动项：

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run*
HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Windows\ run
HKEY_CURRENT_USER\Software\Microsoft\Command Processor
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run*
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\policies\Explorer\run
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\userinit
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon\shell

由于这些自启动项随着计算机的启动而被加载，不易被使用者觉察，病毒和木马正是利用非法修改或添加其中的部分键值传播木马病毒的。

另外，通过“开始”→“程序”→“启动”也可以实现程序自启动。

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\ShellExecuteHooks

这一项也可以被病毒利用。查看注册表中的这些控制键及其子键的键值，是非常有效的木马病毒的检测方法。

再次就是系统服务。“灰鸽子”病毒（Backdoor.Huigezi）就是利用 Windows NT/Windows XP 启动的。目前“灰鸽子”病毒，由于隐蔽性强而有了非常多的变种，甚至专业杀毒软件都无法识别和查杀。在这种情况下，手工查杀就显得非常有效了！

灰鸽子木马分两部分：客户端和服务端。黑客操纵着客户端，利用客户端配置生成出一个服务端程序。服务端文件的名字默认为 G_Server.exe，然后黑客通过各种渠道传播这个木马。可以将它与一张图片绑定；也可以建立一个个人网页，诱骗用户点击；或者利用 IE 漏洞把木马下载到用户的机器上并运行；还可以将文件上传到某个软件下载站点，骗用户下载等。

一旦感染灰鸽子，G_Server.exe 运行后将自己复制到 Windows 安装目录下，释放出

G_Server.dll 和 G_Server_Hook.dll 到 Windows 安装目录下。G_Server.exe、G_Server.dll 和 G_Server_Hook.dll 三个文件相互配合组成了灰鸽子服务端，还有的变种灰鸽子会多释放出一个名为 G_ServerKey.dll 的文件用来记录键盘操作。但是有的变种的 G_Server.exe 这个名称变得不固定，它被重新定制了。例如，当定制服务端文件名为 X.exe 时，生成的文件就是 X.exe、X.dll 和 X_Hook.dll，这就为有效查杀它增加了难度。

Windows 目录下的 G_Server.exe 文件将自己注册成服务，每次开机都能自动运行，同时启动 G_Server.dll 和 G_Server_Hook.dll 并自动退出。G_Server.dll 文件实现后门功能，与控制端客户端进行通信；G_Server_Hook.dll 则通过拦截 API 调用来隐藏病毒。因此，中毒后，我们看不到病毒文件，也看不到病毒注册的服务项。随着灰鸽子服务端文件的设置不同，G_Server_Hook.dll 有时候附在 Explorer.exe 的进程空间中，有时候则是附在所有进程中。

由于灰鸽子拦截了 API 调用，在正常模式下木马程序文件和它注册的服务项均被隐藏，也就是说即使设置了“显示所有隐藏文件”也看不到它们。此外，灰鸽子服务端的文件名也是可以自定义的，这都给手工检测带来了一定的困难。

但是，通过仔细观察能发现，对于灰鸽子的检测仍然是有规律可循的。从上面的运行原理分析可以看出，无论自定义的服务器端文件名是什么，一般都会在操作系统的安装目录下生成一个以 X_hook.dll 结尾的文件。通过这一点，可以准确地手工检测出灰鸽子木马。

由于正常模式下灰鸽子会隐藏自身，因此检测灰鸽子的操作一定要在安全模式下进行。进入安全模式的方法是：启动计算机，在系统进入 Windows 启动画面前，按下 F8 键（或者在启动计算机时按住 Ctrl 键不放），在出现的启动选项菜单中，选择 Safe Mode 或“安全模式”。

由于灰鸽子的文件本身具有隐藏属性，因此又要用到前面讲到（显示隐藏文件）的准备工作了，如图 8-3 所示。

打开 Windows 的“搜索文件”，文件名称输入_hook.dll，搜索位置选择 Windows 的安装目录，如图 8-4 所示。

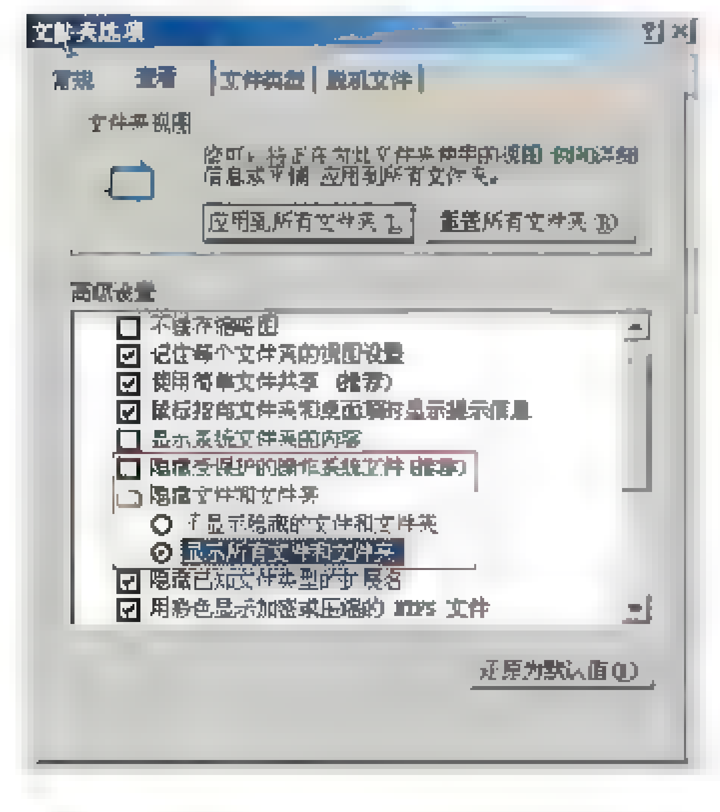


图　8-3

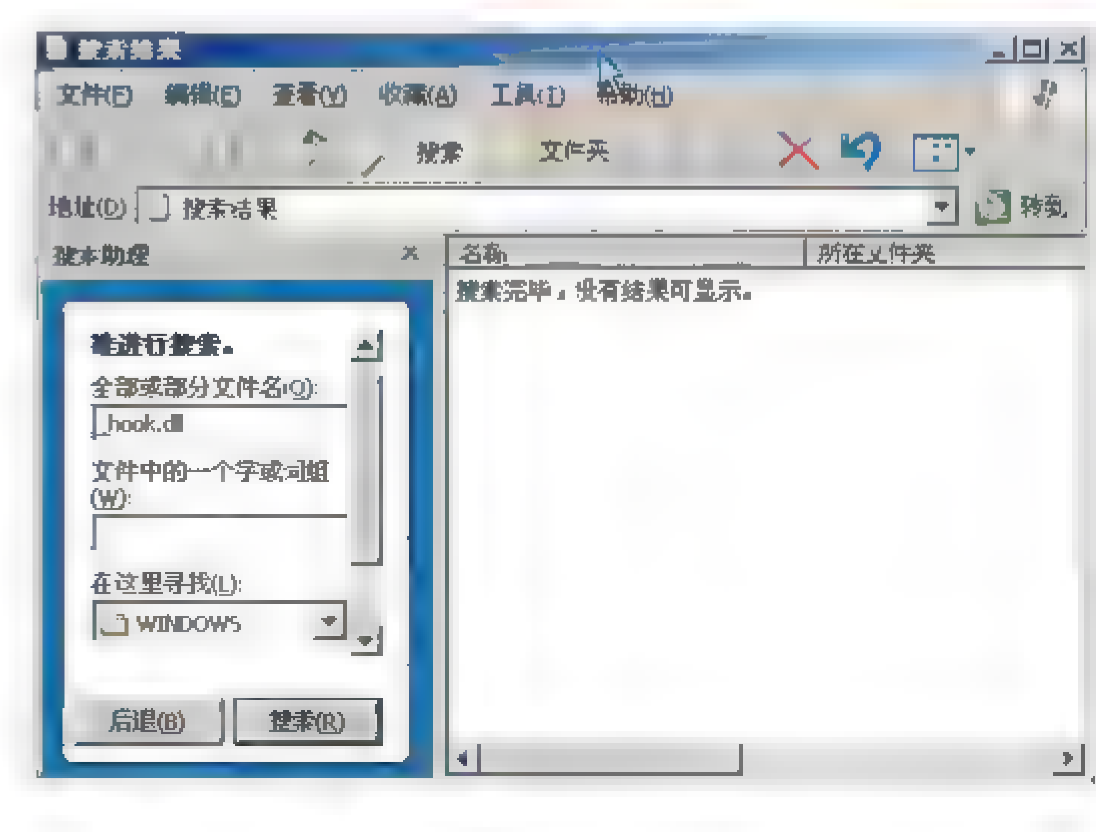


图　8-4

经过搜索，可在 Windows 安装下发现一个名为 Game_Hook.dll 的文件，如图 8-5 所示。根据灰鸽子原理分析可知，如果 Game_Hook.DLL 是灰鸽子的文件，则在 Windows 安

装目录下还会有 Game.exe 和 Game.dll 文件。打开 Windows 目录，果然有这两个文件，同时还有一个用于记录键盘操作的 GameKey.dll 文件，如图 8-6 所示。

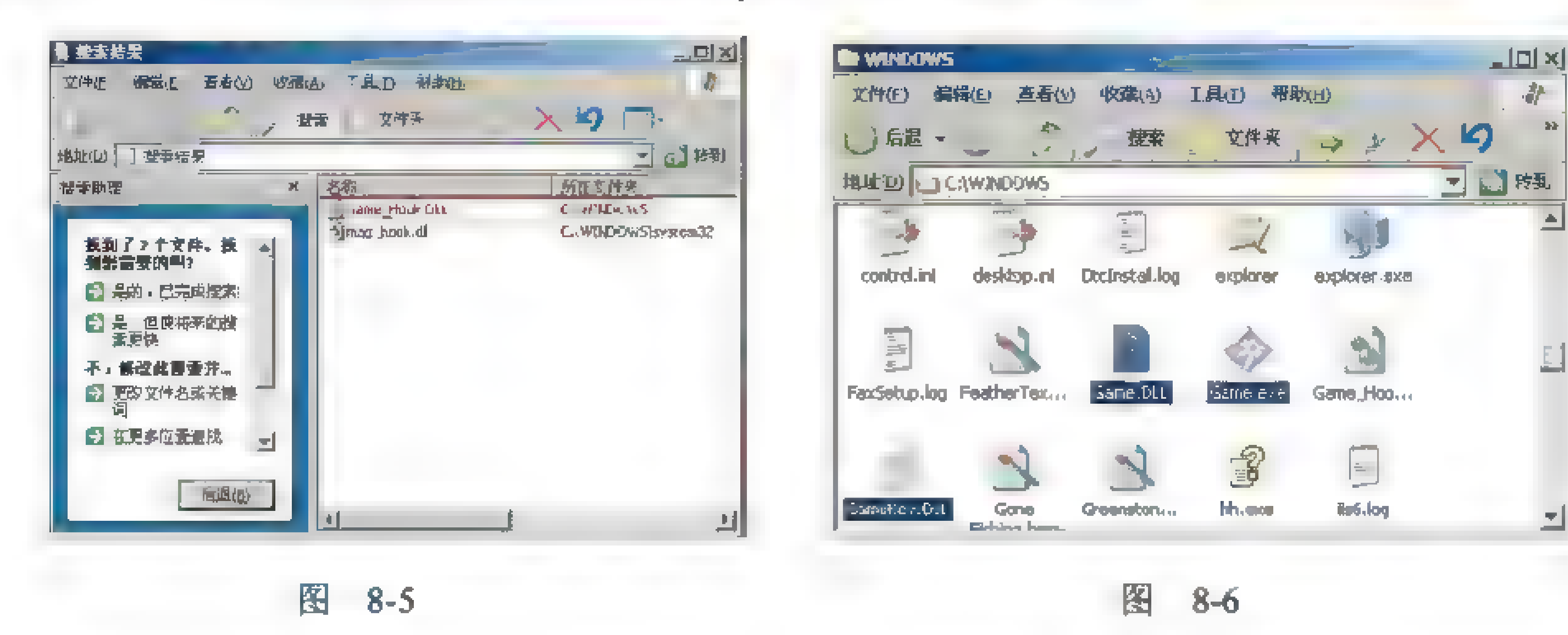


图 8-5　　　　　　图 8-6

由此，可以确定这些文件是灰鸽子木马了，下面就可以进行手动清除。

（1）切换到系统的安全模式。

（2）清除灰鸽子的服务。

打开注册表编辑器（单击“开始”→“运行”命令，输入 Regedit.exe，然后确定），找到 HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services 注册表项。

单击菜单“编辑”→“查找”命令，“查找目标”输入 game.exe，单击“确定”按钮，就可以找到灰鸽子的服务项即 Game_Server，如图 8-7 所示。

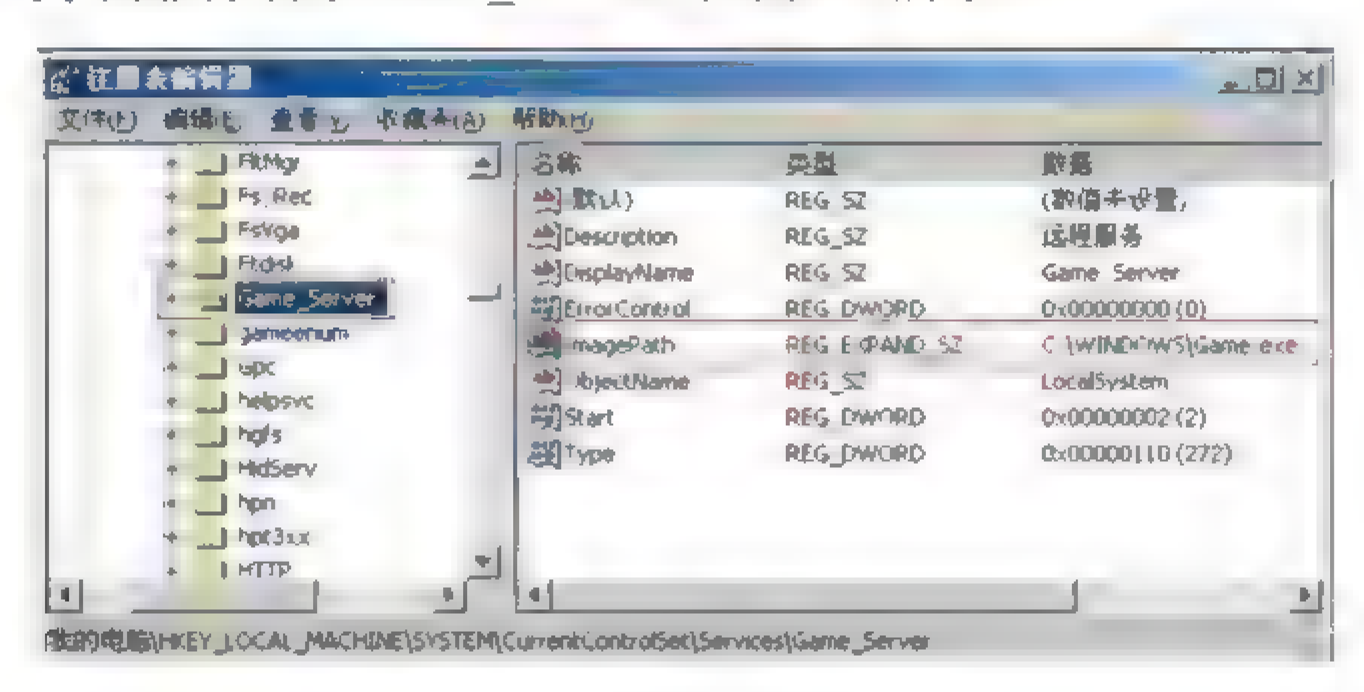


图 8-7

删除整个 Game_Server 项。

（3）删除灰鸽子病毒文件。

在安全模式下删除 Windows 目录下的 Game.exe、Game.dll、Game_Hook.dll 以及 Gamekey.dll 文件，然后重新启动计算机。至此，灰鸽子已经被清除干净。

8.2.3 宏病毒

宏病毒就是使用 Word 的 vba 编程接口编写的具有病毒特征的宏集合，它危害性大。它以二进制文件加密压缩格式存入.doc 或 .dot 文件中，通过 Word 文档或模板进行大量自

我复制及传染。一旦运行宏病毒，相应的 Normal 模板会被传染，所有打开的 Word 文档都会在自动保存时被传染。宏病毒的种类很多，版本也各不相同，为了能查杀各类宏病毒，关键是恢复文件参数。一般的杀毒软件均以查杀宏病毒。

8.2.4　BO 黑洞病毒

BO（Backorifice）黑洞病毒是通过电子邮件进行传播的，它像木马程序一样，有服务器端和客户端程序。它先将 Boserve.exe 服务器端程序植入目标计算机后，BO 程序将目标机信息发回，入侵者通过客户端来控制目标机，Boserve.exe 首先修改目标机的注册表，并自动复制到 System 目录下，将原 Boserve.exe 删除，用自身来替代。用户在使用时，表面上没有发生变化，但实质上，已受到 BO 病毒的控制。

此病毒由 11 组命令组成，它可以搜索服务器端 IP 地址，计算机所有硬件信息、DIR、CD、RD、MD、COPY 和 DELETE 等 DOS 命令从目标机中窃取文件资料和数据等，客户端程序可以像使用自己的计算机一样来控制客户端计算机。用最新的查杀病毒软件能清除 BO 病毒。

8.2.5　邮件病毒

邮件病毒是现代网络发展的不可避免的产物，它利用网络这并不安全的工具大量传播。它的种类很多，给电子邮件用户造成了很大损失。下面以几种流行的邮件病毒为例，让读者全面认识邮件病毒，以便更好地预防。

1. 美丽杀手

此病毒的传染的对象是 Word 类文件，当用户收发这类电子邮件时，交叉感染。一般此病毒有如下表现。

- 注册表：此病毒首先修改注册表，加入如下语句：

HKEY_CURRENT_USERSoftwareMicrosoftOffice "Melissa?"= "…bykwvjibo"。当使用 Word 2000 时，如果 HKEY_CURRENT_USERSoftwareMicrosoftOffice9.0WordSecurity 的 level 的值不为 0，则禁用菜单中的 MACRO/SECURITY。如用 Word 97，则禁用菜单中的 TOOLS/MACRO。

- 利用 VB 建立 OutLook 对象，从全域地址表中获得所有地址，将病毒自动发送到地址中的前 50 个邮箱。它的主题为 ImportanMessageForm-等，所以对于标题不明的邮件，用户最好不要随便打开。
- 当邮件被打开，Word 系统的所有文件将被传染，当系统时钟的时间与日期相同，病毒打开一文件，以此来占用整个系统资源，甚至使系统瘫痪。

2. 求职信病毒

求职信利用微软系统的漏洞，可以自动感染，无需打开附件，破坏力更强，求职信的变种具有很强的隐蔽性，可以随时自动使用不同的邮件主题和内容，同时在邮件内部存放

发送信息的一部分，变种病毒会伪造虚假信息，掩盖病毒的真实来源，能够绕开一些流行杀毒软件的监控，甚至专门针对一些杀毒软件进行攻击，除开可以在网络上利用邮件进行传播外，这些变种病毒还可以利用局域网上的共享文件夹进行传染，因此对于某些不能查杀局域网共享文件病毒的单机版杀毒软件，这将意味着在网络环境下，根本无法彻底清除病毒。对求职信病毒的处理方法如下：

1）安装最新补丁

补丁下载地址：

http://www.microsoft.com/technet/security/bulletin/MS01-020.asp

2）病毒清除

（1）在 Windows 95/98/ME 系统下的清除

先进入 Windows 95/98/ME 系统的安全模式，使用注册表编辑工具 regedit 将网络蠕虫增加的键值删除。

HKEY_LOCAL_MACHINESoftwareMicrosoftWindowsCurrentVersionRun 和 HKEY_LOCAL_MACHINESystemCurrentControlSetServices

要删除的注册表项目是 wink-?.exe 的键值。同时还必须相应地将 Windows 的 SYSTEM 目录下的该随机文件 Wink-?.exe 删除，注意，还必须将回收站清空。删除了相应的病毒文件后，可以重新启动计算机，然后，在 KVW3000、金山毒霸的安装目录下执行 KVD3000.EXE、kavRun.exe 来清除该病毒。

（2）在 Windows 2000/XP 系统下的清除

清除方法基本和 Windows 95/98/ME 系统下的清除方法相同：先以安全模式启动计算机，运行注册表编辑工具，同样删除该网络蠕虫增加的键值：

HKEY_LOCAL_ MACHINESystemCurrentControlSetServices

要删除病毒增加的表项是：wink 开头的随机的表项。必须记住该项目的具体名称（虽然是随机的），然后在系统目录下将该文件删除。注意该文件是隐含的，必须打开显示所有文件的选项才能查看该病毒文件。同样的注册表项还有 HKEY_LOCAL_MACHINE-SoftwareMicrosoftWindowsCurrentVersionRun。

3. 概念病毒

“概念”病毒，也是一种恶性邮件病毒，它通过微软 IE 浏览器解释 Outlook 邮件 MIME 头的漏洞感染和传播，此病毒可以在用户收邮件的时候不知不觉地感染用户的机器，同时利用用户的邮件服务器向外传播。下面介绍对付此种病毒的方法。

1）安装最新的补丁

最新补丁的下载地址是：

http://www.microsoft.com/windows2000/downloads/critical/q269862/default.asp

2）病毒的发现

首先此病毒执行自身复制到 TEMP 目录下，并且修改 Wininit.ini，同时在 Windows 的 System 目录中生成 Load.exe 文件，把 System.ini 中的 Shell explorer.exe 改为 shell - explorer.exe load.exe，从而使病毒每次启动系统时都能激活该病毒，此外，在系统目录中还生成一个副本 riched20.dll，它将覆盖原 Windows 系统的 riched20.dll。因为此病毒是通过邮件进行感染的，它使用 MAPI 函数读取用户的 E-mail 及 SMTP 地址和 E-mail 地址。同

时在临时目录中生成 eml 格式的临时文件，条件成熟后，病毒就用取得的地址将带毒邮件发送出去。

3）利用漏洞检测和其他收邮件系统

可以用其他邮件客户端软件进行代收，不直接进入自己的邮箱，FoxMail 就是一款很好用的邮件客户端收取软件。另外也可以利用漏洞检测软件来完成，达到预防的目的。

可以到 http://www.sky.net.cn 下载天网防火墙，以及到金山的 http://www.iduba.net/download/other/tool_010919_concept.htm 下载工具来补好这些漏洞，从而达到预防邮件病毒入侵的目的。

4）病毒的清除

（1）Windows NT/2000/XP 清除。

- 结束其中进程名称为 X.tmp.exe 和 Load.exe 的进程文件。
- 删除系统 temp 文件夹中文件长度为 57 344 字节的文件。
- 删除系统 System 文件夹中的长度为 57 344 字节的 Riched20.DLL 和 load.exe 文件。
- 打开 System.ini 文件，在[load]中如果有一行"shell=explorer.exe load.exe-dontrunold"，则改为"shell=explorer.exe"。
- 在硬盘区的根目录下寻找 Admin.DLL 文件，如果在根目录下存在该文件，则删除它。
- 打开“控制面板”的用户和密码图标，将 Administrator 组中的 guest 账号删除。
- 把 C 盘的完全共享取消掉。
- 搜索整个硬盘，把所有 readme.eml 的文件删除，这时在没有对系统进行免疫修复前，不要单击任何 readme.eml 文件，按 Ctrl+A 组合键选取全部 readme.eml 文件，删除掉；如果单击了单个 readme.eml 文件，“概念”病毒将利用系统漏洞重新运行。

（2）Win9x 的清除。

- 重启操作系统进入到安全模式。
- 删除系统 temp 文件夹中文件长度为 57 344 字节的文件。
- 删除系统 System 文件夹中的长度为 57 344 字节的 Riched20.DLL 和 load.exe 文件。
- 打开 System.ini 文件，在[load]中如果有一行 shell=explorer.exe load.exe，则改为 shell=explorer.exe。
- 把 C 盘的完全共享取消掉。
- 搜索整个硬盘，把所有 readme.eml 的文件删除，这时在没有对系统进行免疫修复前，请不要单击任何 readme.eml 文件，按 Ctrl+A 组合键选取全部 readme.eml 文件，删除掉；如果单击了单个 readme.eml 文件，“概念”病毒将利用你的系统漏洞重新运行。

8.2.6 CodeRed 病毒

CodeRed 病毒及其变种是一种能够破坏 Windows 2000 的安全体系，修改系统的文件并且安装木马的恶性病毒。现在常见的是红色代码 2 病毒，它用拒绝服务型的入侵方式，把木马程序安装在服务器上，然后用客户端程序来控制木马程序，它传播速度快，对中文操作系统的破坏能力很大，红色代码蠕虫只感染装有 IIS 的服务器，并利用其本身和漏洞

通过网络进行传播。入侵系统后，它将 CMD.EXE 重命名为 root.exe，将其复制到 C 盘和 D 盘的\f1inetpub 和\f1programfilesfiles 目录下。然后，病毒开始扫描整个网络，查寻攻击目标，用存在的安全漏洞植入木马程序，即在 C 盘和 D 盘的根目录下生成一个大小为 8192 字节的 EXPLORER.EXE 木马程序，最后重启系统来执行木马程序。

木马程序修改注册键值，并创建两个虚拟 IIS 目录 C 和 D，分别映射到系统的 C 盘和 D 盘，而这些虚拟目录被赋予读写及可执行权限，这样木马程序通过 IIS 向所有黑客提供了对被感染服务器 C 盘和 D 盘的完全控制能力。然后，木马程序会每 10min 重复进行以上对注册表项的修改。

它的传播速度很快，一旦进入网络，就会在所有拥有 IIS 服务的服务器上传播，发出大量 http 请求，从而使网络进程阻塞，最终导致整个网络瘫痪。

对付此类病毒的方法如下。

1．进行入侵检测

侦测网络流量和 HTTP 服务，找到通过 80 端口发送 HTTP 的 0 字节无用数据包的机器，进而找到已经感染木马程序的 Windows 2000/NT 的机器。

2．安装最新补丁

下载网址：

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/ms01-033.asp，将补丁安装在 Windows 2000/NT 系统上，修补 IIS 系统的漏洞。

3．清除病毒

- 删除 C:.exe 和 D:.exe 文件。
- 修改注册表中被病毒修改的键值 HKLM=0xFFFFFF9D，值改为 0。
- 把 HKLMSVC 中对于 c:\f0 和 d 的完全控制键删除。
- 删除 C 盘及 D 盘\f1inetpub、\f1programfilesfiles 目录中的 root.exe 文件。
- 重启计算机。

8.2.7　熊猫烧香

2006 年下半年肆虐网络的“熊猫烧香”病毒，给人们留下了噩梦般的记忆。它其实是一种蠕虫病毒——尼姆亚 W（Worm.Nimaya.w）的变种，而且是经过多次变种而来的。该病毒还有一个别名叫“武汉男孩”，它是首例造成极大危害的“国产”病毒。由于中毒计算机的可执行文件会出现“熊猫烧香”图案，所以也称为“熊猫烧香”。

1．“熊猫烧香”病毒的危害

（1）被感染的计算机可能会出现蓝屏、频繁重启以及系统硬盘中数据文件被破坏等现象。

（2）感染系统中扩展名为 exe、com、pif、src、html 和 asp 的文件，而且它还能中止

大量的反病毒软件和防火墙软件进程，并且会删除扩展名为 gho 的文件（该文件是当今非常流行的系统备份工具 GHOST 的备份文件）。系统中所有 exe 可执行文件的图标全部被改成熊猫举着三根香的模样。

（3）该病毒可以通过局域网传播，进而感染局域网内所有计算机系统，最终导致企业局域网堵塞甚至瘫痪。

2. “熊猫烧香”病毒的传播途径

（1）在用户的浏览器地址栏添加病毒网址，用户一打开这些网页文件，IE 就会自动连接到指定的病毒网址中下载病毒。

（2）在硬盘各个分区下生成文件 autorun.inf 和 setup.exe，利用 Windows 系统的自动播放功能来运行。

（3）通过 U 盘和移动硬盘等人们防范意识最差的移动存储器进行传播。

（4）还可以通过共享文件夹、系统弱口令等多种方式进行传播。

（5）该病毒会在中毒计算机中所有的网页文件尾部添加病毒代码。一些网站编辑人员的计算机如果被该病毒感染，上传网页到网站后，就会导致用户浏览这些网站时也被病毒感染。

3. “熊猫烧香”病毒的查杀

现在“熊猫烧香”的肆虐似乎已经渐渐成为了历史，如今的杀毒软件已经可以清除大部分的变种，但是病毒和木马对我们的威胁丝毫没有减少。

8.2.8 常见病毒发作日期表

1 月 1 日　BIGBANG
1 月 1 日～9 月 21 日　PLASTIQUE(COBEL)
1 月 5 日　BARROTES
1 月 15 日　CASINO
1 月 25 日　JERUSALEM(JANUARY25TH)
2 月 1 日～2 月 29 日　VIENNA(BETABOYS)
2 月 2 日　DARKAVENGER(AMILIA)
2 月 23 日　SWEDISHBOYS(WHYWINDOWS)
2 月 24 日　SWEDISHBOSS(WHYWINDOWS)
2 月 25 日　SWEDISHBOSS(WHYWINDOWS)
2 月 28 日　ZAPHOD
3 月 1 日～3 月 31 日　FICH
3 月 5 日　X-2(X-1&X-1B)
3 月 6 日　MICHELANGELO
3 月 14 日　ARALE
3 月 15 日　MALTESEAMOEBA

3 月 25 日　MARCH25TH
3 月 31 日～4 月 30 日　MORDOR.1110
4 月 1 日　CASPER
4 月 1 日～4 月 30 日　AKUKU(WIBUR3)
4 月 1 日～6 月 30 日　MONTH4-6
4 月 3 日～12 月 31 日　ITALIANBOY
4 月 12 日　ARCVFRIENDS
4 月 15 日　CASINO
4 月 28 日　ARALE
5 月 1 日～5 月 4 日　1210
5 月 1 日～5 月 31 日　KTHULHU
5 月 5 日　PS-MPC(CINCODEMARYO)
5 月 13 日　ARALE
5 月 17 日　ARALE
6 月 6 日　JERUSALEM(SUB-ZEROB)
6 月 12 日　JUNE12TH
6 月 14 日　GREMLIN
6 月 16 日　JUNE16TH
6 月 17 日～12 月 31 日　JERUSALEM(JUNE17TH)
6 月 26 日　DOSHUNTER
6 月 28 日　CRAZYEDDIE
7 月 1 日～7 月 31 日　ARCV330
7 月 1 日～12 月 31 日　JERUSALEM(MENDOZA)
7 月 4 日　VCL(BEVA96)
7 月 13 日　JULY13TH
7 月 15 日　ARALE
7 月 26 日　JUL26TH
8 月 15 日　CASINO
8 月 16 日　AUGUST16TH
8 月 31 日　BOMBER
9 月 1 日～9 月 30 日　AIRCOP(AIRCOP-B)
9 月 4 日　VIOLATOR(VIOLATORB1)
9 月 8 日　RIP-699
9 月 16 日　IT(VIVAMEXICO)
9 月 20 日～12 月 31 日　PLASTIQUE
9 月 22 日～12 月 31 日　4096
10 月 1 日～12 月 31 日　4096
10 月 4 日　VIOLATOR(VIOLATORB1)
10 月 12 日　JERUSALEM(ANARKIA-B)

10 月 13 日～12 月 31 日　DATACRIME
10 月 15 日　DARKEND
10 月 23 日　KARIN
10 月 28 日　ARAGORN
10 月 30 日　GOTCHA(GOTCHA-MUT4)
10 月 31 日　HALLOWEEN
11 月 1 日　MALTESEAMOEBA
11 月 4 日　VILATOR
11 月 11 日　FLOWER
11 月 12 日　TIMOR
11 月 17 日　NOVEMBER17TH
11 月 17 日～12 月 31 日　NOV17-880
11 月 18 日　TINYVIRUS(KENNEDY)
11 月 24 日　PS-MPC
11 月 30 日　SAMPO
12 月 1 日～12 月 31 日　ANT
12 月 1 日　1253
12 月 4 日　VIOLATORB1
12 月 7 日　VCL
12 月 12 日　ARALE
12 月 19 日～12 月 31 日　FATHER
12 月 19 日～12 月 31 日　CHRISTMAS
12 月 20 日～12 月 25 日　ARCVXMAS
12 月 21 日　POEM
12 月 24 日　ICELANDIC-III
12 月 24 日～12 月 31 日　WITCODE
12 月 24 日～1 月 1 日　CHRISTMASTREE
12 月 25 日　JAPANESECHRISTMAS12 月 28 日 ASH
12 月 31 日　VIOLATORB2

8.3　计算机病毒的防治策略

计算机病毒的防治要从防毒、查毒和解毒三方面来进行，系统对于计算机病毒的实际防治能力和效果也要从防毒能力、查毒能力和解毒能力三方面来评判。

“防毒”是指根据系统特性，采取相应的系统安全措施预防病毒侵入计算机。“查毒”是指对于确定的环境，能够准确地报出病毒名称，该环境包括，内存、文件、引导区（含主引导区）和网络等。“解毒”是指根据不同类型病毒对感染对象的修改，并按照病毒的感染特性所进行的恢复。该恢复过程不能破坏未被病毒修改的内容。感染对象包括内存、引导区（含主引导区）、可执行文件、文档文件和网络等。

防毒能力是指预防病毒侵入计算机系统的能力。通过采取防毒措施，应可以准确地、实时地监测预警经由光盘、软盘和硬盘不同目录之间，局域网、因特网（包括 FTP 方式、E-mail 和 HTTP 方式）或其他形式的文件下载等多种方式进行的传输；能够在病毒侵入系统时发出警报，记录携带病毒的文件，即时清除其中的病毒；对网络而言，能够向网络管理员发送关于病毒入侵的信息，记录病毒入侵的工作站，必要时还要能够注销工作站，隔离病毒源。

查毒能力是指发现和追踪病毒来源的能力。通过查毒应该能准确地发现计算机系统是否感染有病毒，准确查找出病毒的来源，并能给出统计报告；查解病毒的能力应由查毒率和误报率来评判的。

解毒能力是指从感染对象中清除病毒，恢复被病毒感染前的原始信息的能力。解毒能力应用解毒率来评判。

第一代反病毒技术是采取单纯的病毒特征判断，将病毒从带毒文件中清除掉。这种方式可以准确地清除病毒，可靠性很高。后来病毒技术发展了，特别是加密和变形技术的运用，使得这种简单的静态扫描方式失去了作用。随之而来的反病毒技术也向前发展了。

第二代反病毒技术是采用静态广谱特征扫描方法检测病毒，这种方式可以更多地检测出变形病毒，但另一方面误报率也增大，尤其是用这种不严格的特征判定方式去清除病毒带来的风险性很大，容易造成文件和数据的破坏。所以说静态防病毒技术也有难以克服的缺陷。

第三代反病毒技术的主要特点是将静态扫描技术和动态仿真跟踪技术结合起来，将查找病毒和清除病毒合二为一，形成一个整体解决方案，能够全面实现防、查、消等反病毒所必备的各种手段，以驻留内存方式防止病毒的入侵，凡是检测到的病毒都能清除，不会破坏文件和数据。随着病毒数量的增加和新型病毒技术的发展，静态扫描技术将会使查毒软件速度降低，驻留内存防毒模块容易产生误报。

第四代反病毒技术则是针对计算机病毒的发展而基于病毒家族体系的命名规则，基于多位 CRC 校验和扫描机理，启发式智能代码分析模块、动态数据还原模块（能查出隐蔽性极强的压缩加密文件中的病毒）、内存解毒模块和自身免疫模块等先进的解毒技术，较好地解决了以前防毒技术顾此失彼、此消彼长的状态。

另外，现代查杀病毒使用如下技术。

（1）虚拟执行技术：通过虚拟执行方法可以对付加密、变形和压缩型病毒，异型病毒生产机及大部分未知病毒及破坏性病毒，如现在流行的虚拟机 vmware 系列。因为虚拟技术在查杀病毒过程中是在内存中模拟出一个指令执行机器，是在虚拟环境中执行，不影响实际的文件。这种技术是一种很好用的技术。

（2）宏指纹技术：此项技术是基于 Office 复合文档 BIFF 格式精确查杀各类宏病毒的技术，它可以查杀所有的在 Office 文档中存在的可知的和未知的宏病毒，并且可以修复部分被破坏的 Office 文档。

（3）VxD 技术：Windows X 提供的操作系统底层接口技术，Windows X 将操作系统从安全性角度设计为 ring0、ring3 两个级别，其中 Ring0 属于最底层的访问。

此外，对于用户而言，一般对付病毒还应该注意如下几点。

- 不要随意打开来历不明的邮件或附件。

- 安装最新的防毒软件，并随时更新补丁。
- 对常用的存储介质要经常查杀病毒。
- 避免在本机上使用某些共享的系统资源，这些资源可能携带大量病毒。
- 随时定时定期备份数据。

8.4 病毒的检测方法

对系统进行病毒检测，病毒检测的方法有 4 种：特征代码法、校验和法、行为监测法和软件模拟法。

8.4.1 特征代码法

特征代码法早期应用于 SCAN、CPAV 等著名病毒检测工具中。特征代码法是检测已知病毒的最简单、开销最小的方法。

它首先采集已知病毒样本，抽取特征代码。抽取代码时要长度适当，除注意特征代码的独特性，还要注意空间和时间的开销。取各特征代码后，把其归入病毒数据库。如果系统又检测到感染病毒，打开被检测文件，检查文件中是否含有病毒数据库中的病毒特征代码。如果是，则可知病毒的种类及特征。

特征代码法的优点在于检测准确、快速，可识别病毒的名称，误报警率低，依据检测结果，可做解毒处理。但是它不能检测未知病毒、搜集已知病毒的特征代码，费用开销大，在网络上效率低，速度慢。并且不能检查多形性病毒和不能对付隐蔽性病毒。

8.4.2 校验和法

校验和法是将正常文件的内容，计算其校验和，将该校验和写入文件中或写入正常文件中保存。在文件使用过程中，定期地或每次使用文件前，检查文件现在内容算出的校验和与原来保存的校验和是否一致，因而可以发现文件是否感染，这种方法叫校验和法，它既可发现已知病毒又可发现未知病毒。在 SCAN 和 CPAV 工具的后期版本中除了病毒特征代码法之外，还纳入校验和法，以提高其检测能力。

该方法简单，能发现未知病毒，被查文件的细微变化也能发现。但是校验和法对文件内容的变化太敏感，又不能区分正常程序引起的变动，从而频繁报警。用监视文件的校验和来检测病毒，不是最好的方法。另外，校验和法在软件版更新、变更口令、修改运行参数时，都会误报警。同样，校验和法对隐蔽性病毒无效。

8.4.3 行为监测法

利用病毒的特有行为特征性来监测病毒的方法，称为行为监测法。通过对病毒多年的观察、研究，有一些行为是病毒的共同行为，而且比较特殊。在正常程序中，这些行为比较罕见。当程序运行时，监视其行为，如果发现了病毒行为，立即报警。

行为监测法的优点在于可发现未知病毒，可相当准确地预报未知的多数病毒。但是可能误报警，不能识别病毒名称，实现有一定难度。

8.4.4 软件模拟法

多态性病毒每次感染都变化其病毒密码，对付这种病毒，特征代码法失效。因为多态性病毒代码实施密码化，而且每次所用密钥不同，把染毒的病毒代码相互比较，也无法找出相同的可能作为特征的稳定代码。虽然行为检测法可以检测多态性病毒，但是在检测出病毒后，因为不知病毒的种类，难于做消毒处理。

8.5 常用杀毒软件

在病毒和反病毒技术高度对抗的今天，杀毒软件已经成了装机的必备软件。国际和国内的优秀杀毒软件非常多，它们各有各的特点，各有各的长处，但是没有一种杀毒软件可以提供绝对安全的保障。作为用户可以根据自己的需求，结合各种杀毒软件的特点，参考权威专业机构对各个产品的测评选择适合自己的反病毒产品。

下面简单介绍一下当今国际上的优秀反病毒软件。

BitDefender是来自罗马尼亚的老牌杀毒软件，24万超大病毒库，具有功能强大的反病毒引擎及因特网过滤技术。它的功能包括：

（1）永久的防病毒保护。

（2）后台扫描与网络防火墙。

（3）保密控制。

（4）自动快速升级模块。

（5）创建计划任务。

（6）病毒隔离区。

Kaspersky（卡巴斯基）杀毒软件来源于俄罗斯，是世界上最优秀、最顶级的网络杀毒软件，查杀病毒性能远高于同类产品。卡巴斯基杀毒软件具有超强的中心管理和杀毒能力，能真正实现查毒杀毒，提供了一个广泛的抗病毒解决方案。它提供了所有类型的抗病毒防护：抗病毒扫描仪、监控器、行为阻断、完全检验、E-mail通路和防火墙。它支持几乎所有的普通操作系统。卡巴斯基软件控制所有可能的病毒进入端口，它强大的功能和局部灵活性以及网络管理工具为自动信息搜索、中央安装和病毒防护控制提供最大的便利和最少的时间来建构抗病毒分离墙。卡巴斯基抗病毒软件有许多国际研究机构、中立测试实验室和IT出版机构的证书，确认了卡巴斯基软件具有汇集行业最高水准的突出品质。

F-Secure Anti-Virus是来自Linux的故乡芬兰的杀毒软件，集合AVP、LIBRA、ORION和DRACO 4套杀毒引擎，其中一个就是Kaspersky的杀毒内核，而且青出于蓝胜于蓝，该软件采用分布式防火墙技术，对网络流行病毒尤其有效。它集成了多个病毒监测引擎，如果其中一个发生遗漏，就会有另一个去监测。可单一扫描硬盘、一个文件夹或文件，软件更提供密码的保护性，并提供病毒的信息。

PC-cillin是来自台湾的杀毒软件，趋势科技网络安全个人版集个人防火墙、防病毒和

防垃圾邮件等功能于一体，最大限度地提供对桌面机的保护，且并不需要用户进行过多的操作。在用户日常使用及上网浏览时，进行实时的安全防御监控；其内置的防火墙可由用户因地制宜地设定，且专业主控式个人防火墙及木马程序损害清除还原技术的双重保障还可以拒绝各类黑客程序对计算机的访问请求。趋势科技全新研发的病毒阻隔技术，包含主动式防毒应变系统以及病毒扫描逻辑分析技术，不仅能够精准侦测病毒藏匿与化身并予以彻底清除，还能针对特定变种病毒进行封锁与阻隔，让病毒再无可趁之机；强有力的垃圾邮件过滤功能可全面封锁不请自来的垃圾邮件。

国外权威的防病毒软件评测给了 ESET Nod32 很高的分数，在全球共获得超过 40 多个奖项，包括 Virus Bulletin、PC Magazine、ICSA 和 Checkmark 认证等，是全球唯一通过 26 次 VB 100% 测试的防毒软件，高居众产品之榜首。它的产品对 DOS、Windows 9x/Me、Windows NT/XP/2000、Novell Netware Server、Linux 和 BSD 等，都支持。它可以对邮件进行实时监测，占用内存资源较少，清除病毒的速度效果都令人满意。

McAfee VirusScan 是全球最畅销的杀毒软件之一。新版本 McAfee 防毒软件，除了更新操作界面外，还将该公司的 WebScanX 功能整合在一起，增加了许多新功能。除了帮用户侦测和清除病毒，它还有 VShield 自动监视系统，会常驻在 System Tray，当用户从磁盘、网络、E-mail 文件夹中开启文件时便会自动侦测文件的安全性，若文件内含病毒，便会立即警告，并作适当的处理，而且支持鼠标右键的快速选单功能，并可使用密码将个人的设定锁住，让别人无法更改设定。

Norton AntiVirus 是一套强而有力的防毒软件，它可帮用户侦测上万种已知和未知的病毒，并且每当开机时，自动防护便会常驻在 System Tray，当从磁盘、网路和 E-mail 文档中开启档案时便会自动侦测档案的安全性，若档案内含病毒，便会立即警告，并作适当的处理。另外，它还附有 LiveUpdate 的功能，可帮用户自动连上 Symantec 的 FTP Server 下载最新的病毒码，下载完后还自动完成安装更新的动作。

AVG Anti-Virus 是欧洲有名的杀毒软件，AVG Anti-Virus System 功能上相当完整，可即时对任何存取文件进行侦测，防止计算机病毒感染；可对电子邮件和附件进行扫描，防止计算机病毒通过电子邮件和附件传播；病毒资料库里面记录了一些计算机病毒的特性和发作日期等相关资讯；开机保护可在计算机开机时侦测开机型病毒，防止被开机型病毒感染。在扫毒方面，除可扫描磁碟片、硬盘、光盘机外，也可对网络磁碟进行扫描。在扫描时也可只对磁碟片、硬盘和光盘机上的某个目录进行扫描。可扫描文件型病毒、巨集病毒和压缩文件（支持 ZIP、ARJ 和 RAR 等压缩文件即时解压缩扫描）。在扫描时如发现文件感染病毒会将感染病毒的文件隔离至 AVG Virus Vault，待扫描完成后再一并解毒。支持在线升级。现在提供了最新的免费版供用户使用，安装之前先去官方网站填个表，从回信中得到一个序列号。AVG Anti-Virus 有三个版本（专业、服务器和免费），其中有个人非营利使用的免费版本，功能完整，但是某部分功能是无法设定的，例如扫毒排程只能每天一次等。

CA Antivirus 就是反病毒软件 eTrust EZ Antivirus，它已经获得了国际计算机安全协会（International Computer Security Association, ICSA）的认证。ICSA 专门负责检测和认证产品抵御病毒及恶意代码的攻击的有效性。CA 公司表示，在 ICSA 的测试中，CA Antivirus 软件甚至连 In-The-Wild 恶性病毒也可以 100% 地检测出来。新版本采用全新用户界面，更加

易于使用；其新的文件隔离功能可有效防止系统文件被误删；新版本改进了帮助系统，增强了闪动系统托盘图标功能。

Norman Virus Control 是欧洲名牌杀毒软件，为了确保用户的计算机系统得到最好的保护，Norman 数据安全系统提供了多种防毒工具供用户选择，以满足用户的不同需要。此产品结合了先进的病毒扫描引擎、启发式分析技术以及宏验证技术，可有效查杀已知和未知病毒。NVC 可以查杀所有类型的病毒，包括文件和引导扇区病毒而无需使用杀毒软件重新启动开机。

上面介绍的杀毒软件一部分有了汉化版，有的还没有进行汉化，非专业人员使用起来就不太方便，而且病毒库的更新，软件的升级受到一定的网络限制。相比之下，国产的杀毒软件，如瑞星系列、江民的 KV 系列和金山毒霸系列等杀毒软件，为用户提供了更加友好的界面和更加及时的更新服务，而且功能也非常强大。

8.6 计算机病毒的防范技巧

计算机技术的快速发展，使得计算机病毒技术与计算机反病毒技术的对抗越来越尖锐。据统计，现在基本上每天都要出现几十种新病毒，其中很多病毒的破坏性都非常大，稍有不慎，就会给计算机用户造成严重后果。

防患于未然是最好的方法，其实，病毒也只是一段程序，只要注意提高防范意识，做好相应的防护措施，一般情况下是不会受到病毒侵害的。

对于已知的病毒和木马，一般在很短的时间内各大防毒软件商就会提供相应的查杀功能，用户要做的仅仅是及时升级杀毒软件和防火墙软件。

对于未知病毒也不必害怕！通过前面讨论可知，所有的病毒和木马无论其危害大还是小，也无论隐藏手段多么高明，都有共同的特点，即按照“病毒入侵—潜伏并感染—发作”的过程破坏用户的系统。用户所担心的是病毒发作后的破坏后果，实际上在此之前用户有很多机会可以利用有效的方法避免或者把危害减少到最小。具体方法如下。

1. 提高防范意识切断病毒入侵途径

- 时刻保持操作系统获得最新的安全更新，特别是微软的 MS06-014 漏洞，应及时打好该漏洞补丁。
- 提高系统管理员口令的安全级别（采用大小写加特殊符号的组合口令）。
- 采用系统本身的安全策略。
- 不要轻易打开来历不明的文件及邮件程序。
- 对外来的软盘、U 盘和移动硬盘等移动存储介质要随时查毒。
- 不要随意访问来源不明的网站。
- 在网络上下载一切资料时，一定要先做病毒扫描，然后下载。
- 一定要减少机内共享资料。

2. 积累经验及早察觉潜伏的病毒威胁

计算机如果出现 8.1.5 节所列的计算机病毒的破坏现象中的一项或几项，用户就应该

提起注意了，如果升级了杀毒软件之后仍然不能解决问题，那就需要结合 8.2.2 节木马与病毒手工检测的方法进行检测判断，尽早发现病毒隐患才能最大限度地保护系统的安全。

3．运用技术手段查找和判定病毒的存在尽快清除可能的病毒

发现高疑似病毒后，要想清除病毒文件应该做好系统备份，然后运用 8.2.2 节木马与病毒手工清除的方法进行清除。如果清除无效，应该尽快向病毒防护网站报告可疑文件，寻求帮助。

第9章 防火墙安全管理

现代网络安全服务一般有两种，一是存取控制，禁止非法的通信和联网；二是通信安全服务，提供授权数据的完整性、可靠性，具有对同级通信者的访问否定权。当用户连上Internet，就可在中间插入一个或几个中介系统的控制关联，防止通过网络进行的攻击，并提供单一的安全和审计的安装控制点，这些中间系统就是防火墙。由于Internet的开放性，网络安全技术变化很大，Internet的安全技术包括传统的网络安全技术和分布式网络安全技术，主要技术是解决如何利用Internet进行安全通信，同时保护内部网络免受外部攻击。而防火墙正能实现这些技术。

防火墙是一种被动的防御技术，是一类防范措施的总称，是保护计算机网络安全技术性措施之一，是一种隔离控制技术，在内部专用网和外部网络间设置保护，防止对信息资源的非授权访问，防火墙也是目前在网络安全技术中使用最多、最广泛的，因此有必要在网络安全管理中专门来讲述。

9.1 防火墙概述

防火墙（firewall），是现代网络安全技术中最常用的技术，它对内部网络与外部网络（包括Internet等）之间的通信量进行筛选，符合标准的分组将被正常转发，不能通过检查的分组就被丢弃。设置防火墙，就形同装置一个防盗门。一般的防火墙有两个组成部分：两个分组筛选器（路由器）和一个应用程序网关，其示意图如图9-1所示。

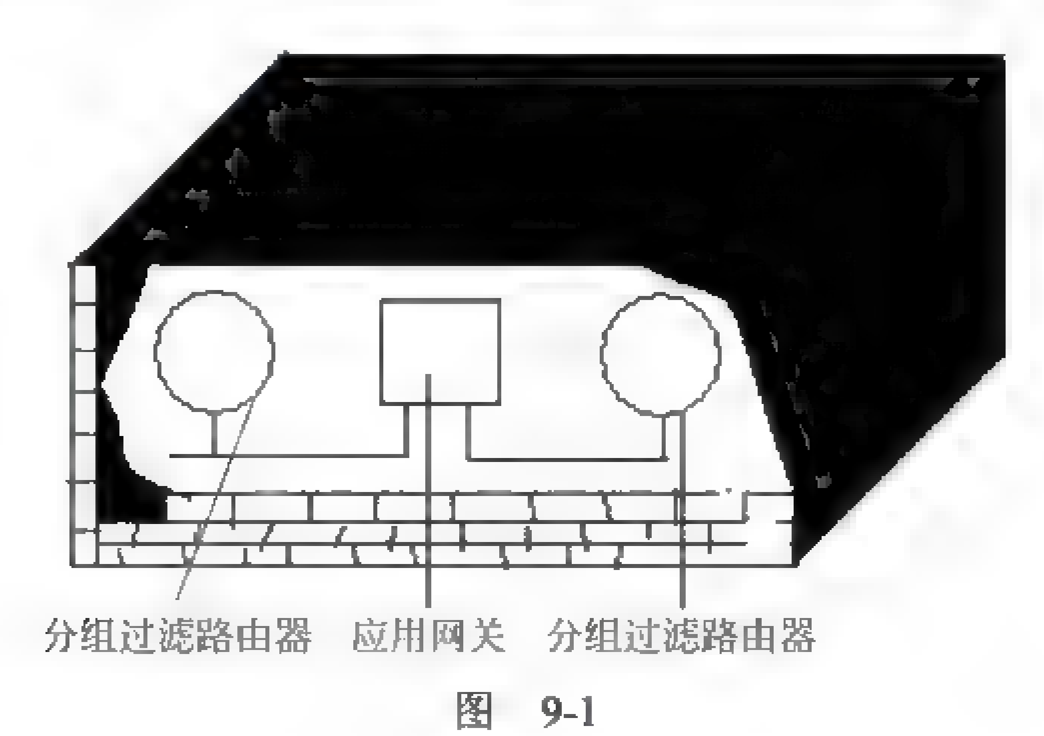


图 9-1

防火墙是用来保护由许多台计算机组成的大型网络，防火墙可以是非常简单的过滤器，也可能是精心配置的应用网关，但它们的原理是一样的，都是监测并过滤所有通向外部网和从外部网传来的信息，防火墙保护着内部敏感的数据不被偷窃和破坏，并用日志记录通信发生的时间和操作。防火墙通常是运行在一台计算机中的软件，它可以识别并屏蔽非法的请求。

防火墙是设置在被保护网络和外部网络之间的一道屏障，以防止发生不可预测的、有潜在破坏性的侵入。它可通过监测、限制和更改跨越防火墙的数据流，尽可能地对外部屏蔽网络内部的信息、结构和运行状况，以此来实现网络的安全保护。

9.1.1　防火墙的特点

防火墙已经成为网络的首选，它的广泛应用与它的特点有紧密的联系，防火墙具有如下特点：

- 广泛的服务支持。防火墙动态地将应用层过滤能力和认证相结合，可以实现应用层的大部分服务（如 WWW 服务、HTTP 服务和 FTP 服务等）。
- 数据的加密支持。保证通过 Internet 进行虚拟专用网络和电子商务不受损坏。
- 防电子欺骗。欺骗是从外部获取网络访问权的常用手段，它使数据包好似来自网络内部。防电子欺骗功能是保证数据包的 IP 地址与网关接口相符，防止通过修改 IP 地址的方法进行非授权访问。防火墙能监视过滤数据包，欺骗的数据包将被过滤。
- 过滤不安全服务和非法用户。
- 控制对特殊站点的访问。
- 提供了监视 Internet 安全和预警的方便端点。
- 防火墙能强化安全策略。
- 防火墙能有效地记录 Internet 上的活动；作为访问的唯一点，防火墙能在被保护的网络和外部网络之间进行记录。
- 防火墙限制暴露用户点（所以现在一般使用应用代理服务器）。

9.1.2　实现防火墙的技术

实现防火墙的主要技术有数据包过滤、应用网关、代理服务、IP 通道、隔离域名服务器和网络地址转换器等相关技术，分别介绍如下。

1．包过滤技术（Packet Filter）

包过滤是在网络层中对数据包进行有选择的通过。依据系统内事先设定的过滤逻辑，检查数据流中每个数据包后，根据数据包的源地址、目的地址、所用的 TCP 端口与 TCP 链路状态等因素来确定是否允许数据包通过。

包过滤是在 IP 层实现的，因此，它可以只用路由器完成。包过滤根据包的源 IP 地址、目的 IP 地址、源端口、目的端口及报文传递方向等报头信息来判断是否允许报文通过。现在也出现了一种可以分析报文数据区内容的智能型包过滤器。包过滤器的应用非常广泛，因为 CPU 用来处理包过滤的时间可以忽略不计。而且这种防护措施对用户透明，合法用户

在进出网络时，根本感觉不到它的存在，使用起来很方便。

包过滤有一个很关键的弱点是不能在用户级别上进行过滤，即不能识别不同的用户和防止 IP 地址的盗用。如果攻击者把自己主机的 IP 地址设成一个合法主机的 IP 地址，就可以很轻易地通过报文过滤器。

包过滤技术作为防火墙的应用有三类：

- 路由设备在完成路由选择的数据转发之外，同时进行包过滤，这是目前较常用的方式。
- 在工作站上使用软件进行包过滤，但是此方式价格较贵。
- 在一种称为屏蔽路由器的路由设备上启动包过滤功能。

2．应用网关技术（Application Gateway）

应用网关技术是利用网络应用层上的协议过滤。它针对特别的网络应用服务协议即数据过滤协议，并且能够对数据包分析并形成相关的报告。应用网关对某些易于登录和控制所有输入输出的通信的环境给予严格控制，以防有价值的程序和数据被窃取。在实际工作中，应用网关一般由专用工作站系统来完成。

3．代理服务（Proxy Server）

代理服务是设置在 Internet 防火墙网关的专用应用级代码。这种代理服务准许网管员允许或拒绝特定的应用程序或一个应用的特定功能。包过滤技术和应用网关是通过特定的逻辑判断来决定是否允许特定的数据包通过，一旦判断条件满足，防火墙内部网络的结构和运行状态便“暴露”在外来用户面前，这就引入了代理服务的概念，即防火墙内外计算机系统应用层的“链接”由两个终止于代理服务的“链接”来实现，这就成功地实现了防火墙内外计算机系统的隔离。同时，代理服务还可用于实施较强的数据流监控、过滤、记录和报告等功能。代理服务技术主要通过专用计算机硬件（如工作站）来承担。代理服务器则是代表网络内部用户的代理者，它实际上是一个应用层上的网关。当用户使用 TCP/IP 应用时，给 Proxy（代理）提供合法身份和授权信息，Proxy 就和被访问主机联系，并在两个通信点之间中继传递 IP 数据包。IP 包处理的过程对用户是透明的。代理服务器一般包括以下几种。

（1）应用代理服务器（Application Gateway Proxy）

这种防火墙是在网络应用层提供授权检查及代理服务。例如，当外部某台主机试图访问（如 Telnet）受保护网时，它必须先在防火墙上经过身份认证。通过身份认证后，防火墙运行一个专门为 Telnet 设计的程序，把外部主机与内部主机连接。在这个过程中，防火墙可以限制用户访问的主机、访问时间及访问的方式。同样，受保护网络内部用户访问外部网时也须先登录到防火墙上，通过验证后，才可使用 Telnet 或 FTP 等有效命令。

应用网关代理的优点是既可以隐藏内部 IP 地址，也可以给单个用户授权，即使攻击者盗用了一个合法的 IP 地址，他也通不过严格的身份认证。因此应用网关比报文过滤具有更高的安全性。但是这种认证使得应用网关不透明，用户每次连接都要接受验证，这给用户带来许多不便，而且这种代理技术需要为每个应用写专门的程序。

（2）回路级代理服务器

也就是通常所说的“一般”代理服务器，它适用于多个协议，但它不能解释应用协议，

需要通过其他方式来获得信息，所以，回路级代理服务器通常要求修改过的用户程序。

套接字服务器（Sockets Server）就是回路级代理服务器。套接字（Sockets）是一种网络应用层的国际标准。当受保护网络客户机需要与外部网交互信息时，在防火墙上的套接字服务器检查客户的 UserID、IP 源地址和 IP 目的地址，经过确认后，套接字服务器才与外部的服务器建立连接。对用户来说，受保护网与外部网的信息交换是透明的，感觉不到防火墙的存在，那是因为网络用户不需要登录到防火墙上。但是客户端的应用软件必须支持 Sockets fired API（套接字层防火墙应用程序接口），受保护网络用户访问公共网所使用的 IP 地址也都是防火墙的 IP 地址。

4. IP 通道（IP Tunnels）

当两个相关的网络相隔很远，要通过 Internet 通信的情况下，可以采用 IP Tunnels 来防止 Internet 上的入侵者截取信息，实质是建立虚拟专用网。试分析一下其工作原理。

子网 A 中一主机（IP 地址为 X.X.X.X）欲向子网 B 中某主机（IP 地址为 Y.Y.Y.Y）发送报文，该报文经过本网防火墙 FW1（IP 地址 N.N.N.1）时，防火墙判断该报文是否发往子网 B，若是，则再增加一报头，变成从此防火墙到子网 B 防火墙 FW2（N.N.N.2）的 IP 报文，而将原 IP 地址封装在数据区内，同原数据一起加密后经 Internet 发往 FW2。FW2 接收到报文后，若发现源 IP 地址是 FW1 的，则去掉附加报头，解密，在本网上传送。从 Internet 上看，就只是两个防火墙的通信。即使黑客伪装了从 FW1 发往 FW2 的报文，由于 FW2 在去掉报头后不能解密，会抛弃报文。

5. 网络地址转换器（NAT Network Address Translate）

当受保护网连到 Internet 上时，受保护网用户若要访问 Internet，必须使用一个合法的 IP 地址。但由于合法 Internet IP 地址有限，而且受保护网络往往有自己的一套 IP 地址规划（非正式 IP 地址）。网络地址转换器就是在防火墙上装一个合法 IP 地址集。当内部某一用户要访问 Internet 时，防火墙动态地从地址集中选一个未分配的地址分配给该用户，该用户即可使用这个合法地址进行通信。同时，对于内部的某些服务器如 Web 服务器，网络地址转换器允许为其分配一个固定的合法地址。外部网络的用户可通过防火墙来访问内部的服务器。这种技术既缓解了少量的 IP 地址和大量的主机之间的矛盾，又对外隐藏了内部主机的 IP 地址，提高了安全性。

6. 隔离域名服务器（Split Domain Name Server）

这种技术是通过防火墙将受保护网络的域名服务器与外部网的域名服务器隔离，使外部网的域名服务器只能看到防火墙的 IP 地址，无法了解受保护网络的具体情况，这样可以保证受保护网络的 IP 地址不被外部网络知悉。

9.2　防火墙的类型

总的来讲防火墙的类型有如下几种，网络级防火墙（包括包过滤防火墙）、应用级网关、电路级防火墙以及状态监视器等。

9.2.1 网络级防火墙

网络级防火墙是基于源地址和目的地址、应用或协议以及每个 IP 包的端口来作出通过与否的判断。一个路由器便是一个“传统”的网络级防火墙，大多数的路由器都能通过检查这些信息来决定是否将所收到的包转发，但它不能判断出一个 IP 包来自何方，去向何处。

网络级防火墙可以判断这一点，它可以提供内部信息以说明所通过的连接状态和一些数据流的内容，把判断的信息同规则表进行比较，在规则表中定义了各种规则来表明是否同意或拒绝包的通过。包过滤防火墙检查每一条规则直至发现包中的信息与某规则相符。如果没有一条规则能符合，防火墙就会使用默认规则，一般情况下，默认规则就是要求防火墙丢弃该包。另外，通过定义基于 TCP 或 UDP 数据包的端口号，防火墙能够判断是否允许建立特定的连接，如 Telnet、FTP 连接。

网络级防火墙的访问控制规则举例：

- 允许网络 202.206.197.0 使用 FTP（端口号为 21）访问主机 202.206.197.1。
- 允许 IP 地址为 202.103.1.18 和 202.103.1.14 的用户 Telnet（端口号为 23）到主机 202.206.197.2 上。
- 允许任何地址的 E-mail（端口号为 25）进入主机 202.206.197.3。
- 允许任何 WWW 数据（端口号为 80）通过。
- 不允许其他数据包进入。

网络级防火墙简洁、速度快、费用低，并且对用户透明，但是对网络的保护很有限，因为它只检查地址和端口，对网络更高协议层的信息无理解能力。在网络级防火墙中使用最典型的是数据包过滤防火墙。

数据包过滤防火墙

数据包过滤（Packet Filtering）技术是在网络层对数据包进行选择，如图 9-2 所示，选择的依据是系统内设置的过滤逻辑，称为访问控制表（Access Control Table）。通过检查数据流中每个数据包的源地址、目的地址、所用的端口号和协议状态等因素，或它们的组合来确定是否允许该数据包通过。

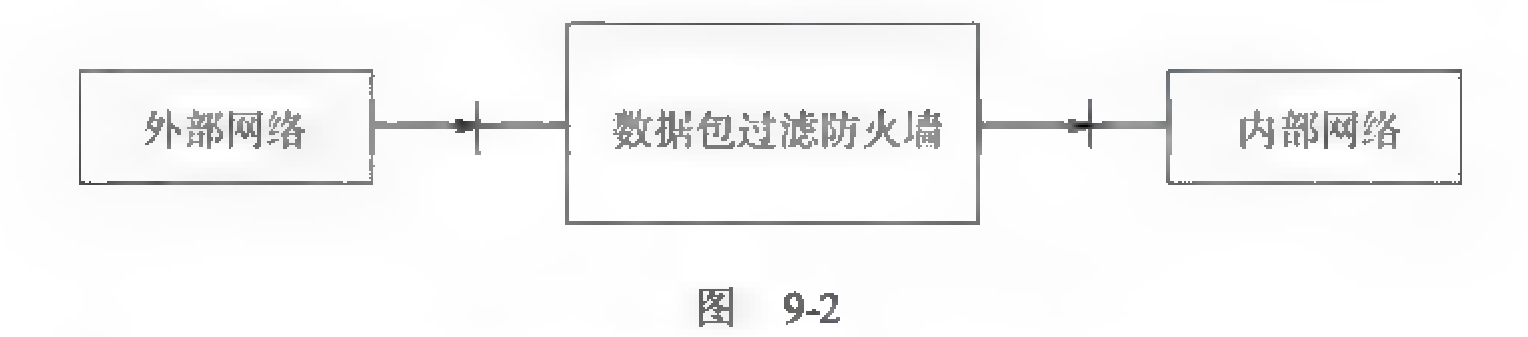


图 9-2

在 Internet 上都使用数据包交换数据，网络上的所有信息都按照一定的协议规则分割成不同大小的数据包，数据包中含有发送者的 IP 地址和接收者的 IP 地址。数据包在 Internet 上传递时，路由器会根据数据包的 IP，在路由表中选择一条路径把数据包发送到目的 IP，数据包过滤防火墙会自动检查所有通过数据包的 IP 地址，按照此防火墙的过滤规则对数据包进行过滤，如果防火墙认定这个 IP 不适合，防火墙会自动屏蔽掉这类数据包。

数据包过滤防火墙一般作为第一道网络防护的防线，它速度快并且对于用户是透明的，但是如果网络只是用单一的数据包过滤防火墙来防护，那是很危险的。因为入侵者将用信息包冲击以及同步淹没等手段来攻击这样的防火墙，造成整个内部网络的暴露及服务器的死锁。

9.2.2　应用级网关防火墙

应用级网关防火墙你也许不熟悉，它的别名是代理服务器。这种防火墙有较好的访问控制，是目前最安全的防火墙技术，但它对用户是不透明的，用户在受信任的网络上通过防火墙访问 Internet 时，经常会发现存在延迟并且必须进行多次登录（login）才能访问 Internet 或内联网的问题。应用级防火墙应用于特定的 Internet 服务，如 HTTP、NNTP、FTP 和 Telnet 等。代理服务器通常运行在两个网络之间，它对于客户来说就像是一台真的服务器一样，而对于外界的服务器来说，它又是一台客户机。当代理服务器接受到用户的请求后，会检查用户请求的站点是否符合要求，如果允许用户访问该站点，代理服务器会去那个站点取回所需信息再转发给客户。代理服务器通常都拥有一个高速缓存，这个缓存内有用户经常访问站点的内容，在下一个用户要访问同样的站点时，服务器就用不着重复地去抓同样的内容，既节约了时间也节约了网络资源。代理服务器会像一堵真的墙那样挡在内部用户和外界之间，从外面只能看到代理服务器而看不到任何的内部资源，诸如用户的 IP 等。应用级网关比单一的包过滤更为可靠，而且会详细地记录下所有的访问记录。但是应用级网关的访问速度慢，因为它不允许用户直接访问网络。而且应用级网关需要对每一个特定的互联网服务安装相应的代理服务软件，用户不能使用未被服务器支持的服务，它的效率不如网络级防火墙。

常用的应用级防火墙的相应代理服务器，例如 HTTP、NNTP、FTP、Telnet、rlogin 和 X-window 等，但是，对于新开发的应用，尚没有相应的代理服务，它们将通过网络级防火墙和一般的代理服务。

9.2.3　电路级网关防火墙

电路级网关用来监控受信任的客户或服务器与不受信任的主机间的 TCP 握手信息，这样来决定该会话（session）是否合法。电路级网关在 OSI 模型中会话层上过滤数据包，这样比包过滤防火墙要高二层。

实际上电路级网关并非作为一个独立的产品存在，它与其他的应用级网关结合在一起，如 Trust Information Systems 公司的 Gauntlet Internet Firewall。DEC 公司的 Alta Vista Firewall 等产品。另外，电路级网关还提供一个重要的安全功能——代理服务器（Proxy Server）。代理服务器是个防火墙，在其上运行一个叫做“地址转移”的进程，将所有公司内部的 IP 地址映射到一个“安全”的 IP 地址，这个地址是由防火墙使用的。但是，作为电路级网关也存在着一些缺陷，因为该网关是在会话层工作的，它无法检查应用层级的数据包。

9.2.4 规则检查防火墙

规则检查防火墙结合了包过滤防火墙、电路级网关和应用级网关的特点。同包过滤防火墙一样，规则检查防火墙能够在OSI网络层上通过IP地址和端口号，过滤进出的数据包。它也像电路级网关一样，能够检查SYN和ACK标记和序列数字是否逻辑有序。当然它也像应用级网关一样，可以在OSI应用层上检查数据包的内容，查看这些内容是否能符合公司网络的安全规则。

规则检查防火墙虽然集成前三者的特点，但是不同于一个应用级网关的是，它并不打破客户/服务器（C/S）模式来分析应用层的数据，它允许受信任的客户机和不受信任的主机建立直接连接。规则检查防火墙不依靠与应用层有关的代理，而是依靠某种算法来识别进出的应用层数据，这些算法通过已知合法数据包的模式来比较进出数据包，这样从理论上就能比应用级代理在过滤数据包上更有效。

目前在市场上流行的防火墙大多属于规则检查防火墙，因为该防火墙对于用户透明，在OSI最高层上加密数据，不需要去修改客户端的程序，也不需对每个需要在防火墙上运行的服务额外增加一个代理。例如OnTechnology软件公司生产的OnGuard和CheckPoint软件公司生产的FireWall-1防火墙，都是一种规则检查防火墙。

未来的防火墙将位于网络级防火墙和应用级防火墙之间，也就是说，网络级防火墙将变得更加能够识别通过的信息，而应用级防火墙在目前的功能上则向“透明”、“低级”方向发展。最终防火墙将成为一个快速注册稽查系统，可保护数据以加密方式通过，使所有组织可以放心地在节点间传送数据。

9.2.5 状态监视器

状态防火墙的安全特性是非常好的，它采用了一个在网关上执行网络安全策略的软件引擎，称之为检测模块。检测模块在不影响网络正常工作的前提下，采用抽取相关数据的方法对网络通信的各层实施监测，抽取部分数据（即状态信息），并动态地保存起来作为以后制定安全决策的参考。检测模块支持多种协议和应用程序，并可以很容易地实现应用和服务的扩充。与其他安全方案不同，当用户访问到达网关的操作系统前，状态监视器（Stateful Inspection）要抽取有关数据进行分析，结合网络配置和安全规定做出接纳、拒绝、鉴定或给该通信加密等决定。一旦某个访问违反安全规定，安全报警器就会拒绝该访问，并作记录，向系统管理器报告网络状态。

状态监视器的另一个优点是它会监测RPC（Remote Procedure Call）和UDP（User Datagram Protocol）之类的端口信息。包过滤和代理网关都不支持此类端口。这种防火墙无疑是非常坚固的，但它的配置非常复杂，而且会降低网络的速度。

总而言之，无论是什么类型的防火墙，都只是一层安全的防护，防火墙的配置与安全管理是最重要的，你可以为自己的网络设置多层防火墙，即使一层防火墙被突破，网络还可以由其他防火墙来保护。但是防火墙不是万能的，网络的整体安全的部署才是最重要的，防火墙只是第一道保护屏障。

9.3　防火墙体系结构

防火墙体系结构主要有三种：

- 双重宿主主机体系结构。
- 被屏蔽主机体系结构。
- 被屏蔽子网体系结构。

9.3.1　双重宿主主机体系结构

双重宿主主机体系结构是围绕具有双重宿主的主机计算机而构筑的，该计算机至少有两个网络接口。这样的主机可以充当与这些接口相连的网络之间的路由器，它能够从一个网络到另一个网络发送 IP 数据包。实现双重宿主主机的防火墙体系结构禁止这种发送功能，所以 IP 数据包从一个网络并不是直接发送到其他网络。防火墙内部的系统能与双重宿主主机通信，同时防火墙外部的系统能与双重宿主主机通信，但是这些系统不能直接互相通信，它们之间的 IP 通信被完全阻止。

双重宿主主机的防火墙体系结构是相当简单的，双重宿主主机位于两者之间，并且被连接到 Internet 和内部的网络。在双重宿体主机体系中应用最广泛的是双穴主机网关，这种网关是用一台装有两块网卡的堡垒主机作防火墙；两块网卡各自与受保护网和外部网相连。堡垒主机上运行着防火墙软件，可以转发应用程序，提供服务等。

9.3.2　屏蔽主机体系结构

屏蔽主机体系结构使用一个单独的路由器提供来自仅仅与内部的网络相连的主机的服务。在这种体系结构中，主要的安全由数据包过滤。

在屏蔽的路由器上的数据包过滤按以下方法设置：堡垒主机是 Internet 上的主机能连接到内部网络上的系统的桥梁。仅有某些确定类型的连接被允许。任何外部的系统试图访问内部的系统或者服务将必须连接到这台堡垒主机上。堡垒主机需要拥有高等级的安全。在屏蔽路由器中数据包过滤配置的可选方案如下：

- 允许其他的内部主机为了某些服务与 Internet 上的主机连接。
- 不允许来自内部主机的所有连接。
- 用户可以针对不同的服务混合使用上述手段，某些服务可以被允许直接经由数据包过滤，而其他服务可以只允许间接地经过代理。这完全取决于用户实行的安全策略。

9.3.3　屏蔽子网体系结构

屏蔽子网体系结构添加额外的安全层到被屏蔽主机体系结构，即通过添加周边网络更进一步地把内部网络与 Internet 隔离开。

堡垒主机是用户的网络上最容易受侵袭的计算机。任凭用户尽最大的力气去保护它，

它仍是最有可能被侵袭的计算机，因为它本质上是能够被侵袭的计算机。如果在屏蔽主机体系结构中，用户的内部网络对来自用户的堡垒主机的侵袭门户洞开，那么用户的堡垒主机是非常诱人的攻击目标。在它与用户的其他内部计算机之间没有其他的防御手段时，如果有人成功地侵入屏蔽主机体系结构中的堡垒主机，那就毫无阻挡地进入了内部系统。

通过在周边网络上隔离堡垒主机，能减少在堡垒主机上侵入的影响。屏蔽子网体系结构的最简单的形式为：两个屏蔽路由器，每一个都连接到周边网。其中一个位于周边网与内部的网络之间，另一个位于周边网与外部网络之间。为了侵入用这种类型的体系结构构筑的内部网络，入侵者必须通过两个路由器。即使侵袭者设法侵入堡垒主机，他仍然必须通过内部路由器。在此情况下，没有损害内部网络的单一的易受侵袭点。作为入侵者，只是进行了一次访问。

1. 周边网络

周边网络是另一个安全层，是在外部网络与用户的被保护的内部网络之间的附加的网络。如果侵袭者成功地侵入用户的防火墙的外层领域，周边网络在那个侵袭者与用户的内部系统之间提供一个附加的保护层。

在许多网络设置中，用给定网络上的任何机器来查看这个网络上的每一台机器的通信是可能的。对局域网中所使用的技术，入侵者都很熟悉，入侵者可以通过查看那些在Telnet、FTP及rlogin会话期间使用过的口令，成功地探测出口令及电子邮件用以监视网络等。

对于周边网络，如果入侵周边网上的堡垒主机，入侵者只能探听到周边网上的通信。因为所有周边网上的通信来自或者通往堡垒主机或Internet。

因为没有严格的内部通信能越过周边网，所以，如果堡垒主机被损害，内部的通信仍将是安全的。总的来说，在堡垒主机或者外部的通信，仍然是可监视的。防火墙设计工作的一部分就是确保这种通信不至于机密到阅读它将损害站点的完整性。

2. 堡垒主机

在屏蔽的子网体系结构中，用户把堡垒主机连接到周边网，这台主机便是接受来自外界连接的主要入口。

- 对于进来的电子邮件（SMTP）会话，传送电子邮件到站点。
- 对于进来的FTP连接，转接到站点的匿名FTP服务器。
- 对于进来的域名服务（DNS）站点查询等。
- 在外部和内部的路由器上设置数据包过滤来允许内部的客户端直接访问外部的服务器。
- 设置代理服务器在堡垒主机上运行（如果用户的防火墙使用代理软件），来允许内部的客户端间接地访问外部的服务器。用户也可以设置数据包过滤来允许内部的客户端在堡垒主机上同代理服务器交谈，但是禁止内部的客户端与外部世界之间直接通信（即拨号入网方式）。

3. 内部路由器

内部路由器（也称阻塞路由器）保护内部的网络，使之免受Internet和周边网络的侵

犯。内部路由器为用户的防火墙执行大部分的数据包过滤工作。它允许从内部网络到 Internet 的有选择的出站服务。这些服务是用户的站点能使用数据包过滤，而不是代理服务安全支持和安全提供的服务。

内部路由器所允许的在堡垒主机和用户的内部网络之间服务，可以不同于内部路由器所允许的在 Internet 和用户的内部网络之间的服务。限制堡垒主机和内部网络之间服务的理由，是减少由此而导致的受到来自堡垒主机侵袭的机器的数量。

4. 外部路由器

外部路由器（又称访问路由器）保护周边网络和内部网络使之免受来自 Internet 的侵犯。实际上，外部路由器倾向于允许几乎任何东西从周边网络出站，并且它们通常只执行非常少的数据包过滤。保护内部计算机的数据包过滤规则在内部路由器和外部路由器上基本上是一样的，如果在规则中有允许侵袭者访问的错误，错误就可能出现在两个路由器上。

外部路由器由外部群组提供，同时用户对它的访问被限制。外部群组可能愿意放入一些通用型数据包过滤规则来维护路由器，但是不愿意使维护复杂或者使用频繁变化的规则组。

外部路由器能有效地执行的安全任务，是阻止从 Internet 上伪造源地址进来的任何数据包。这样的数据包自称来自内部的网络，但实际上是来自 Internet。

5. 被屏蔽子网

在内部网络和外部网络之间建立一个被隔离的子网，用两台分组过滤路由器将这一子网分别与内部网络和外部网络分开。两个分组过滤路由器放在子网的两端，在子网内构成一个“非军事区”DMZ，内部网络和外部网络均可访问被屏蔽子网，但禁止它们穿过被屏蔽子网通信，如 WWW 和 FTP 服务器可放在 DMZ 中。有的屏蔽子网中还设有一堡垒主机作为唯一可访问点，支持终端交互或作为应用网关代理。这种配置的危险带仅包括堡垒主机、子网主机及所有连接内网、外网和屏蔽子网的路由器。

如果入侵者试图完全破坏防火墙，则重新配置连接三个网的路由器，既不切断连接又不要把自己锁在外面，同时又不使自己被发现，这样也还是可能的。但若禁止网络访问路由器或只允许内网中的某些主机访问它，则攻击会变得很困难。在这种情况下，攻击者必须先侵入堡垒主机，然后进入内网主机，再返回来破坏屏蔽路由器，整个过程中不能引发警报。

9.3.4　防火墙体系结构的组合形式

建造防火墙时，一般很少采用单一的技术，通常是多种解决不同问题的技术组合。这种组合主要取决于网管中心向用户提供什么样的服务，以及网管中心能接受什么等级风险。采用哪种技术主要取决于经费，投资的大小或技术人员的技术、时间等因素。一般有以下几种形式：

- 使用多堡垒主机。
- 合并内部路由器与外部路由器。

- 合并堡垒主机与外部路由器。
- 合并堡垒主机与内部路由器。
- 使用多台内部路由器。
- 使用多台外部路由器。
- 使用多个周边网络。
- 使用双重宿主主机与屏蔽子网。

9.4 防火墙的选择

在规划网络时，就必须考虑整体网络的安全性，而在这其中，防火墙是第一道防护。防火墙产品很多，如何进行防火墙的选择，是一个必须考虑的问题，建议考虑以下几点。

1. 好的防火墙是一个整体网络的保护者

好的防火墙是整体网络的保护者，它所保护的是整个局域网。

2. 好的防火墙必须能弥补其他操作系统的不足

好的防火墙必须是建立在操作系统之前而不是在操作系统之上，所以操作系统的漏洞并不会影响到防火墙系统所提供的安全性，由于硬件平台的普及以及执行效率的因素，大部分企业均会把对外提供各种服务的服务器分散至许多操作平台上，但我们在无法保证所有主机安全的情况下，选择防火墙作为整体安全的保护者，这正说明了操作系统提供了 B 级或是 C 级的安全并不一定会直接对整体安全造成影响，好的防火墙必须能弥补操作系统的不足。

3. 好的防火墙应该为使用者提供不同平台的选择

由于防火墙并非完全由硬件构成，所以软件（操作系统）所提供的功能以及执行效率一定会影响到整体的表现，而使用者的操作意愿及熟悉程度也是必须考虑的重点。因此一个好的防火墙不但本身要有良好的执行效率，也应该提供多平台的执行方式供使用者选择，毕竟使用者才是完全的控制者，应该选择一套符合现有环境需求的软件，并非为了软件的限制而改变现有环境。

4. 好的防火墙应能向使用者提供完善的售后服务

由于有新的产品出现，就有人会研究新的破解方法，所以一个好的防火墙提供者就必须有一个庞大的组织作为使用者的安全后盾，也应该有众多的使用者所建立的口碑为防火墙作证。

5. 有提供完整的安全检查功能

好的防火墙应该向使用者提供完整的安全检查功能，但是一个安全的网络仍必须依靠使用者的观察及改进。

6. 好的防火墙还能实现 IP 转换

IP 转换能隐藏内部网络真正的 IP，使入侵者无法直接入侵内部网，另外节省的 IP 可作为内部使用。

7. 好的防火墙应该有双重 DNS

当内部网络使用没有注册的 IP 地址或是防火墙进行 IP 转换时，DNS 也必须经过转换，同样的一个主机在内部的 IP 与给予外界的 IP 将会不同，所以双重 DNS 防火墙是很必要的。

8. 查杀病毒功能

大部分防火墙都可以与防病毒防火墙搭配实现查杀病毒功能，有的防火墙则可以直接集成扫毒功能，有的是杀毒由防火墙完成，有的是由专用的计算机完成。

防火墙也是网络上的主机，除了配置防火墙的控制连接以及它自身的服务配置、端口设置以外，还应该考虑到防火墙自身的安全，因为它自身的安全性决定着内部网络的安全性。防火墙大部分都安装在网络操作系统上，如 Linux、Windows 系列等，在防火墙主机上执行的除了防火墙软件外，所有的程序与系统核心，也大多来自操作系统本身的原有程序。当防火墙上所执行的软件出现安全漏洞时，防火墙本身也将受到威胁。此时，任何的防火墙控制机制都可能失效，因为当入侵者取得了防火墙上的控制权以后，入侵者可以通过防火墙入侵内部网络，所以防火墙自身要具有相当高的安全保护。在其他章节详细地讲述了各个网络操作系统的安全配置及管理，读者可以在使用相应操作系统的时候作为参考。

9.5　常用防火墙的配置与管理

防火墙配置一般有三种方式：Dual-homed、Screened-host 和 Screened-subnet。

Dual-homed 方式最简单。Dual-homedGateway 放置在两个网络之间，这个 Dual-homed Gateway 又称为 bastionhost。这种结构成本低，但是它有单点失败的问题。这种结构没有增加网络安全的自我防卫能力，并且是入侵者的首选目标，它极易被攻破。一旦被攻破，整个网络也就暴露了，所以这种配置方式不利于安全的管理。

Screened-host 方式中的 Screeningrouter 为保护 Bastionhost 的安全建立了一道屏障。它将所有进入的信息先送往 Bastionhost，并且只接收来自 Bastionhost 的数据作为出去的数据。这种结构依赖 Screeningrouter 和 Bastionhost，只要有一个失败，整个网络就暴露了。

Screened-subnet 方式包含两个 Screeningrouter 和两个 Bastionhost。在公共网络和私有网络之间构成了一个隔离网，称之为“停火区（Demilitarized Zone，DMZ）”，Bastionhost 放置在“停火区”内。这种结构安全性好，只有当两个安全单元被破坏后，网络才被暴露，但是成本也很昂贵。

下面来介绍常见的防火墙的配置。

9.5.1　配置防火墙

现在来介绍基于 iptables 的防火墙。这个产品和以前的 ipchains 一样，是 Linux 下标准的 IP 包过滤工具。技术上，它是 ipchains 的下一代产品，增加了一些重要的功能，如 QoS、带宽控制，同时也进一步细化了过滤规则的控制。

1．iptables 语法介绍

iptables 的 IP 过滤规则由链和对应的处理规则组成，防火墙的规则指定所检查包的特征、目标。若包不匹配则送往该链下一条规则检查；若匹配，则由目标值确定下一条规则。这些目标值可以是：ACCEPT（通过）、DROP（删除）、QUEUE（排队）和 RETURN（返回）。一旦到达了目标，链就结束了。

iptables 默认定义了几个标准链，即 INPUT（输入链）、OUTPUT（输出链）、FORWARD（转发链）、PREROUTING 和 POSTROUTING。如果你曾经使用过 ipchains，那么要注意，iptables 的 INPUT 和 OUTPUT 的定义与 ipchains 不同，现在 INPUT 和 OUTPUT 只包含事实上进入/离开本机的包，过去的 forward 包穿过三条链的情况不再出现了。相当于以前的 INPUT 链的链名是 PREROUTING，也就是一切通过本机 NIC 进入的都要通过 PREROUTING，但事实上只有进入本机的才通过 INPUT。类似地，所有通过本机 NIC 发出的数据都要通过 POSTROUTING，但只有本机发出的数据包才通过 OUTPUT，如图 9-3 所示。

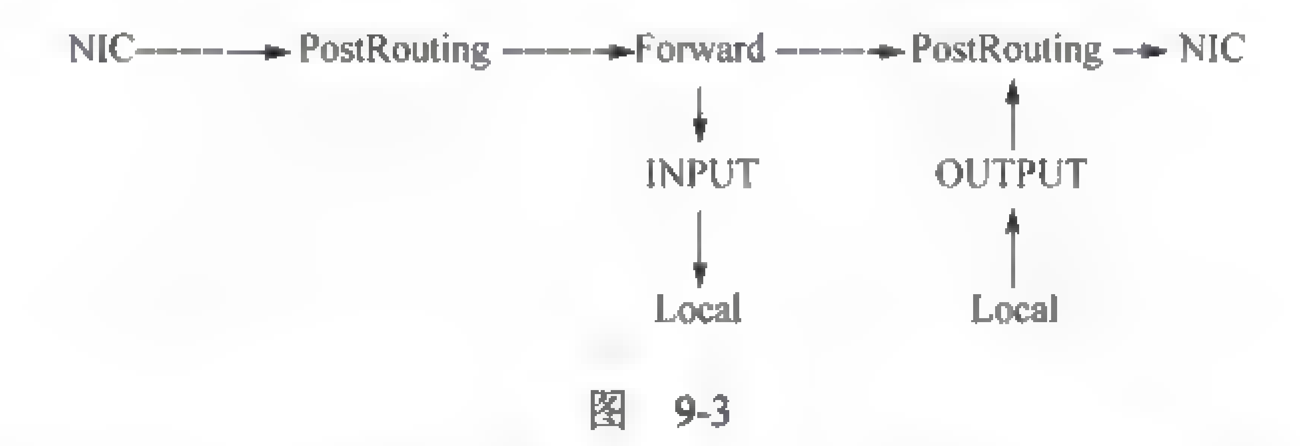


图　9-3

iptables 中的另外一个主要概念是表，用-t，--table 来表示。简单地说，表是一个满足特定功能的链的分组，系统用它来管理默认规则，例如，要执行 Nat，内核必须含有 Nat 的表项设定，iptables 定义了三个表：Filter、Nat 和 Mangle。其含义如表 9-1 所示。

表 9-1　iptables 表名

表　名	链　名	描　述
Filter	INPUT	进入本机的数据包
	FORWORD	通过本机转发的包
	OUTPUT	由本机发出去的包
Nat	PREROUTING	网络地址转换到来的包
	OUTPUT	网络地址转换路由之前本地的包
	POSTROUTING	网络地址转换出去的包
Mangle	PREROUTING	变更路由之前进入的包
	OUTPUT	变更路由之前出去的包

如果内核已经定义了需要的表，那么就可以用 iptables 命令来管理既定的规则，其主要的命令选项如表 9-2 所示。

表 9-2　iptables 命令选项

选　项	描　述
-A, --append	向所选链追加规则
-D --delete	从所选链删除规则，可以指定序号或者指定为要匹配的规则
-R --replace	替代所选链规则
-I --insert	以指定规则序向所选链插入规则
-L --list	列于所选链的所有规则
-F --flush	清空所选链
-Z --zero	清空所有链的包及字节的计数器
-N --new-chain	根据给定名称建立新的用户定义链
-X --delete-chain	删除指定的用户定义链
-P --policy	设置给定目标链规则
-E --rename-chain	根据用户给出的名字重命名指定链
-h	帮助

规则中可以使用的参数，如表 9-3 所示。

表 9-3　iptables 参数

选　项	描　述
-p, --protocol[!] protocol	指定检查的协议，可以是 TCP、UDP 和 ICMP 中的一种或全部，可以是数值，也可以是/etc/protocols 中定义的协议名，协议名前加“!”表示相反的规则
-s, --source[!] address[/mask]	指定源地址，可以是主机名、网络名和简单 IP 地址
-d, --destination[!] address[/mask]	指定目标地址
-j, --jump target	目标跳转
-i, --in-interface[!][name]	接收包的接口名称
-o, --out-interface[!][name]	发送包的接口名称
[!]-f, --fragment	在分片的包中，规则只询问第二及以后的片
-c，--set-counters PKTS BYTES	初始化包、字节规则

除此之外，还有扩展参数如表 9-4 所示。

表 9-4　iptables 扩展参数

名　称	选　项	描　述
tcp	--source-port[!][port[:port]]	指定源端口或端口范围
	--destination-port[!][port[:port]]	指定目标端口或端口范围
	--tcp-flags[!]mask comp	匹配指定的 TCP 标记
	[!]--syn	只匹配设置 SYN 位而清除 ACK 和 FIN 位的 TCP 包
	--tcp-option[!] number	若 TCP 选项设置则匹配
udp	--source-port[!][port[:port]]	指定源端口或端口范围
	--destination-port[!][port[:port]]	指定目标端口或端口范围
icmp	--icmp-type[!] typename	指定 ICMP 类型

续表

名　称	选　　项	描　　述
mac	--mac-source[!] address	匹配物理地址
limit	--limit rate	最大平均匹配速率
	--limit-burst number	待匹配包最大初始值

iptables 还预定义了一些特殊目标扩展，它们可以和普通的目标一起使用，对包提供特殊的处理，如表 9-5 所示。

表 9-5　iptables 特殊目标扩展

名　　称	选　　项	描　　述
LOG	--log-level level	设置记录级别
	--log-prefix prefix	在记录信息前加特定前缀
	--log-tcp-sequence	记录 TCP 序列号
	--log-tcp-options	记录来自 TCP 包头部的选项
	--log-ip-options	记录来自 IP 包头部的选项
MARK	--set-mark mark	设置包的 netfilter 标记值
REJECT	--reject-with type	禁止数据包通过，同时返回相应的 ICMP 错误信息
SNAT	--to-source<ipaddr>[-<ipaddr>] [:port-port]	网络源地址转换
DNAT	--to-destination<ipaddr>[-<ipaddr>][:port-port]	网络目的地址转换
MASQUERADE	--to-ports<port>[-<port>]	IP 伪装，参数指定使用的源端口范围，覆盖默认的 SNAT 源地址选择
REDIRECT	--to-ports<port>[-<port>]	将本来应该穿过本机的目标数据包转发到本地端口，参数指定使用的本地端口或端口范围

2．配置实例

下面给出了一个 iptables 的脚本模型，其含义不难分析，通常这样的脚本放在 /etc/rc.d/init.d，使用 rcX.d 调用。

需要注意的是，如果进行了 Nat，则相应的 DNS 配置文件和 qmail/control 文件要修改添加 masq 后的主机名。

```
#!/bin/sh
#make me executable(chmod a+x rc.firewall) and run me on boot (add in
rc.l ocal)
#iptables firewall script

#interface definitions
internet_IFACE=eth0
intranet_IFACE=eth1
extranet_IFACE=eth2
intranet_ADDR=202.206.196.16/28
extranet_ADDR=192.168.0.0/24
```

```
MASQ SERVER=202.206.196.216
internet SERVER=202.206.196.17
intranet_SERVER=202.206.196.18
extranet_SERVER=192.168.0.2
#testing
#set -x
#we need proxy arp for the intranet network
echo 0 > /proc/sys/net/ipv4/conf/eth0/proxy_arp
echo 0 > /proc/sys/net/ipv4/conf/eth1/proxy_arp
echo 1 > /proc/sys/net/ipv4/conf/eth2/proxy_arp
#turn on ip forwarding
echo 1 > /proc/sys/net/ipv4/ip_forward
#turn on antispoofing protection
for f in/proc/sys/net/ipv4/conf/*/rp_filter; do echo 1 > $ f; done

#flush built in rules
iptables -F INPUT
iptables -F OUTPUT
iptables -F FORWARD
iptables -F PREROUTING -t nat
iptables -F OUTPUT -t nat
iptables -F POSTROUTING -t nat
#deny everything for now
#iptables -A INPUT -j DROP
#iptables -A FORWARD -j DROP
#iptables -A OUTPUT -j DROP

#set which addresses jump to which chains
iptables -P FORWARD ACCEPT

iptables -A INPUT -d 192.168.0.0/24 -j DROP
iptables -A INPUT -s 192.168.0.0/24 -d 192.168.0.0/24 -j ACCEPT
iptables -A INPUT -s 192.168.0.0/24 -d 202.206.196.16/28 -j ACCEPT
iptables -A INPUT -d 202.206.196.16/28 -j DROP
iptables -A INPUT -s 202.206.196.16/28 -d 202.206.196.16/28 -j ACCEPT
iptables -A INPUT -s 202.206.196.16/28 -d 192.168.0.0/24 -j ACCEPT
iptables -A INPUT -j LOG --log-prefix "input"
iptables -A OUTPUT -j ACCEPT
iptables -A OUTPUT -j LOG --log-prefix "output"
```

9.5.2　防火墙的管理

防火墙的管理应该注意如下几点。

1. 防火墙的失效状态

防火墙能否起到安全防护作用，不仅要看它工作是否正常，能否阻挡或捕捉到恶意攻击和非法访问的蛛丝马迹，而且要看到一旦防火墙被攻破，它的状态如何。按级别来分，它有以下 4 种状态：

- 未受伤害能够继续正常工作。
- 关闭并重新启动，同时恢复到正常工作状态。
- 关闭并禁止所有的数据通行。
- 关闭并允许所有的数据通行。

前两种状态比较理想，而第四种最不安全。但是许多防火墙由于无条件进行失效状态测试和验证，无法确定其失效状态等级，因此网络存在安全隐患。

2. 防火墙的动态维护

防火墙安装和投入使用后，并非万事大吉。要想充分发挥它的安全防护作用，必须对它进行跟踪和维护，要与商家保持密切的联系，时刻注视商家的动态。因为商家一旦发现其产品存在安全漏洞，那么会尽快发布补救（Patch）产品，此时应尽快确认真伪（防止特洛伊木马等病毒），并对防火墙软件进行更新。

3. 防火墙的非法访问

深入分析研究防火墙技术，利用防火墙配置和实现的漏洞，可以对它实施攻击。通常情况下，有效的攻击都是从相关的子网进行的，因为这些网址得到了防火墙的信赖，虽说成功与否尚取决于机遇等其他因素，但对攻击者而言很值得一试。

下面以数据包过滤防火墙为例，简要描述可能的攻击过程。

这种类型的防火墙以 IP 地址作为鉴别数据包是否允许其通过的条件，而这恰恰是实施攻击的突破口。许多防火墙软件无法识别数据包到底来自哪个网络接口，因此攻击者无须表明进攻数据包的真正来源，只需伪装 IP 地址，取得目标的信任，使其认为来自网络内部即可。IP 地址欺骗攻击正是基于这类防火墙对 IP 地址缺乏识别和验证的机制。

通常主机 A 与主机 B 的 TCP 连接（中间有或无防火墙）是通过主机 A 向主机 B 提出请求建立起来的，而其间 A 和 B 的确认仅仅根据由主机 A 产生并经主机 B 验证的初始序列号 ISN。具体分三个步骤：

（1）主机 A 产生它的 ISN，传送给主机 B，请求建立连接。

（2）B 接收到来自 A 的带有 SYN 标志的 ISN 后，将自己本身的 ISN 连同应答信息 ACK 一同返回给 A。

（3）A 再将 B 传送来的 ISN 及应答信息 ACK 返回给 B。

这样，正常情况，主机 A 与 B 的 TCP 连接就建立起来了。

IP 地址欺骗攻击的第一步是切断可信赖主机。这样可以使用 TCP 淹没攻击，使得信赖主机处于“自顾不暇”的忙碌状态，相当于被切断，这时目标主机会认为信赖主机出现了故障，只能发出无法建立连接的 RST 包而无暇顾及其他。

攻击者最关心的是猜测目标主机的 ISN。为此，可以利用 SMTP 的端口（25），通常

它是开放的，邮件能够通过这个端口，与目标主机打开（Open）一个 TCP 连接，因而得到它的 ISN。在此有效期间，重复这一过程若干次，以便能够猜测和确定 ISN 的产生和变化规律，这样就可以使用被切断的可信赖主机的 IP 地址向目标主机发出连接请求。请求发出后，目标主机会认为它是 TCP 连接的请求者，从而给信赖主机发送响应（包括 SYN），而信赖主机目前仍忙于处理 Flood 淹没攻击产生的“合法”请求，因此目标主机不能得到来自信赖主机的响应。

现在攻击者发出回答响应，并连同预测的目标主机的 ISN 一同发给目标主机。

随着不断地纠正预测的 ISN，攻击者最终会与目标主机建立一个会晤。通过这种方式，攻击者以合法用户的身份登录到目标主机而不需进一步确认。如果反复试验使得目标主机能够接收对网络的 root 登录，那么就可以完全控制整个网络。

4．防火墙安全的几个威胁

防火墙安全防护面临威胁的主要原因如下。

- SOCK 的错误配置。
- 不适当的安全政策。
- 强力攻击。
- 允许匿名的 FTP 协议。
- 允许 TFTP 协议。
- 允许 rlogin 命令。
- 允许 X-window 或 OpenWindows。
- 端口映射。
- 可加载的 NFS 协议。
- 允许 Win95/NT 文件共享。

9.5.3　华为的 VRP3 防火墙配置

通用路由平台（Versatile Router Platform，VRP）是整个 VRP 平台的核心，它实现了 OSPF、BGP、IS-IS、RIP、EIGRP、PIM DM/SM 等多种单播和多播路由协议，支持路由迭代、路由策略和路由聚合等丰富的路由特性，VRP 中的防火墙主要是指基于访问控制列表（ACL）的包过滤、基于应用层的包过滤防火墙 ASPF 和地址转换。

1．ACL/包过滤防火墙

ACL/包过滤应用在路由器中，就为路由器增加了对数据包的过滤功能。ACL/包过滤实现对 IP 数据包的过滤，对路由器需要转发的数据包，先获取数据包的包头信息，包括 IP 层所承载的上层协议的协议号，数据包的源地址、目的地址、源端口和目的端口等，然后和设定 ACL 规则进行比较，根据比较的结果决定对数据包进行转发或者丢弃。ACL/包过滤提供了对分片报文检测过滤的支持。包过滤防火墙将检测报文类型有非分片报文、首片分片报文和非首片分片报文；获得报文的三层（IP 层）信息（基本 ACL 规则和不含三层以外信息的高级 ACL 规则）及三层以外的信息（包含三层以外信息的高级 ACL 规则）用

于匹配，并获得配置的ACL规则。对于配置了精确匹配过滤方式的高级ACL规则，包过滤防火墙需要记录每一个首片分片的三层以外的信息，当后续分片到达时，使用这些保存的信息对ACL规则的每一个匹配条件进行精确匹配。应用精确匹配过滤后，包过滤防火墙的执行效率会略微降低，配置的匹配项目越多，效率降低越多，可以配置门限值来限制防火墙最大处理的数目。

ACL/包过滤防火墙配置主要需要配置步骤如下。

（1）允许或禁止防火墙。在系统视图输入操作命令：

firewall enable

如果是禁止防火墙，输入 undo firewall enable。

系统默认情况下禁止防火墙。

（2）设置防火墙默认过滤方式。在系统视图输入操作命令：

firewall default permit

如果设置默认过滤方式为禁止通过，输入 firewall default deny。

在防火墙开启时，系统默认为允许。

（3）设置包过滤防火墙分片报文检测开关。在系统视图中输入操作命令。

打开分片报文检测开关 firewall fragments-inspect。

如果需要关闭分片报文检测开关，输入 undo firewall fragments-inspect。

注意：只有打开了分片报文检测开关，精确匹配模式才能真正有效。

（4）配置分片报文检测的上、下门限值，在系统视图输入操作命令：

firewall fragments-inspect { high | low } { default | number }

如果恢复上限分片状态记录数目为默认值，输入 undo firewall fragments-inspect { high | low }。

注意：默认的上限（high）分片状态记录数目为 2000；下限（low）分片状态记录数目为 1500。

（5）在接口上应用访问控制列表，在接口视图输入操作命令：

firewall packet-filter { acl-number | acl-name } { inbound | outbound } [match-fragments { normally | exactly }]

如果取消接口上过滤接收报文的规则，输入 undo firewall packet-filter { acl-number | acl-name } { inbound | outbound }

（6）包过滤防火墙显示与调试。

在完成上述配置后，在所有视图下执行如下 display 命令可以显示包过滤防火墙的运行情况，通过查看显示信息验证配置的效果。执行如下 debugging 命令可以对包过滤防火墙进行调试。

display firewall-statistics { all | interface interface-name | fragments-inspect }
#显示接口的有关防火墙的统计信息

debugging firewall { all | icmp | tcp | udp | others } [interface interface-name]
#打开防火墙包过滤调试信息开关

undo debugging firewall { all | icmp | tcp | udp | others } [interface interface-name]
#关闭防火墙包过滤调试信息开关

下面通过一个公司配置防火墙的实例来说明防火墙的配置。

该公司通过一台 Quidway 路由器的接口 Serial1/0/0 访问 Internet，路由器与内部网通过以太网接口 Ethernet0/0/0 连接。公司内部对外提供 WWW、FTP 和 Telnet 服务，公司内部子网为 129.38.1.0，其中，内部 FTP 服务器地址为 129.38.1.1，内部 Telnet 服务器地址为 129.38.1.2，内部 WWW 服务器地址为 129.38.1.3，公司对外地址为 202.38.160.1。在路由器上配置了地址转换，这样内部 PC 可以访问 Internet，外部 PC 可以访问内部服务器。通过配置防火墙，希望实现以下要求：

- 外部网络只有特定用户可以访问内部服务器。
- 内部网络只有特定主机可以访问外部网络。
- 假定外部特定用户的 IP 地址为 202.39.2.3。

具体的配置步骤如下：

```
# 在路由器 Quidway 上允许防火墙
[Quidway] firewall enable
# 设置防火墙默认过滤方式为允许包通过
[Quidway] firewall default permit
# 创建访问控制列表 101
[Quidway] acl number 101
# 配置规则禁止所有 IP 包通过
[Quidway-acl-adv-101] rule deny ip
# 配置规则允许特定主机访问外部网，允许内部服务器访问外部网
[Quidway-acl-adv-101] rule permit ip source 129.38.1.4 0
[Quidway-acl-adv-101] rule permit ip source 129.38.1.1 0
[Quidway-acl-adv-101] rule permit ip source 129.38.1.2 0
[Quidway-acl-adv-101] rule permit ip source 129.38.1.3 0
# 创建访问控制列表
[Quidway] acl number 102
# 配置规则允许特定用户从外部网访问内部服务器
[Quidway-acl-adv-102] rule permit tcp source 202.39.2.3 0 destination 202.38.160.1 0
#配置规则允许特定用户从外部网取得数据（只允许端口大于 1024 的包）
[Quidway-acl-adv-102] rule permit tcp destination 202.38.160.10 0 destination-port gt 1024
# 将规则 101 作用于从接口 Ethernet0/0/0 进入的包
[Quidway-Ethernet0/0/0] firewall packet-filter 101 inbound
# 将规则 102 作用于从接口 Serial1/0/0 进入的包
[Quidway-Serial1/0/0] firewall packet-filter 102 inbound
```

2. ASPF 的配置与实例

ASPF（Application Specific Packet Filter）是针对应用层的包过滤，即基于状态的报文过滤。它和普通的静态防火墙协同工作，以便于实施内部网络的安全策略。ASPF 能够检测试图通过防火墙的应用层协议会话信息，阻止不符合规则的数据报文穿过。为保护网络安全，基于访问控制列表的包过滤可以在网络层和传输层检测数据包，防止非法入侵。ASPF

能够检测应用层协议的信息，并对应用的流量进行监控。同时能针对 DoS 进行检测和防范。使用 Java Blocking（Java 阻断）来保护网络不受有害的 Java Applets 的破坏。它还支持端口到应用的映射，用于应用层协议提供的服务使用非通用端口时的情况。它增强的会话日志功能。可以对所有的连接进行记录，包括：记录连接的时间、源地址、目的地址、使用的端口和传输的字节数。ASPF 对应用层的协议信息进行检测，并维护会话的状态，检查会话的报文的协议和端口号等信息，阻止恶意的入侵。ASPF 能对如下的协议：FTP、HTTP、SMTP、RSTP、H.323、TCP 和 UDP 的流量进行监测。

ASPF 配置中需要允许防火墙使用，同时配置访问控制列表，然后定义 个 ASPF 策略，最后在选定的接口上应用。

下面介绍一下如何定义一个 ASPF 策略。

（1）创建一个 ASPF 策略。在系统视图下输入。

aspf-policy aspf-policy-number

如果删除创建一个 ASPF 策略，输入 undo aspf-policy aspf-policy-numbe，其中 aspf-policy-number 为 ASPF 策略号，范围为 1～99。

（2）配置空闲超时值。输入 aging-time { syn | fin | tcp | udp } seconds。

如果恢复默认的空闲超时值，输入 undo aging-time { syn | fin | tcp | udp }。

该任务用来配置 TCP 的 SYN 状态等待超时值、FIN 状态等待超时值，TCP 和 UDP 会话表项空闲状态超时值。默认情况 syn、fin、tcp、udp 的超时时间分别为 30s、5s、3600s 和 30s。

（3）配置应用层协议检测。输入。

detect protocol [aging-time seconds]

如果要删除配置的应用协议检测，输入 undo detect protocol。

应用层协议 protocol 可取值 ftp、h323、smtp、rtsp、http。在 protocol 选择 http 时，可以配置 Java 阻断，命令为：

detect http { java-list acl-number } [aging-time seconds]

如果取消对 HTTP 的检测规则，输入 undo detect http。

（4）配置一般 TCP 和 UDP 检测。在 ASPF 策略视图下键入。

detect tcp [aging-time seconds]

配置通用 TCP 协议检测

detect udp [aging-time seconds]

配置通用 UDP 协议检测

undo detect tcp

删除通用 TCP 协议检测

undo detect udp

删除通用 UDP 协议检测

（5）在接口上应用，在接口视图下，输入如下命令。

firewall aspf aspf-policy-number { inbound | outbound }

如果删除该接口上应用的 ASPF 策略，输入

undo firewall aspf aspf-policy-number{ inbound | outbound }

（6）ASPF 显示与调试。

在完成上述配置后，在所有视图下执行如下 display 命令可以显示 ASPF 的运行情况，通过查看显示信息验证配置的效果。在用户视图下执行 debugging 命令查看 ASPF 调试信息。

display aspf all

显示所有 ASPF 配置情况

display aspf interface

显示应用 ASPF 策略和访问列表的接口配置

display aspf policy aspf-policy-number

显示一个特定 ASPF 策略的配置

display aspf session

显示 ASPF 当前会话状态

debugging aspf { all | detail | events | ftp | h323 | http | rtsp | session | smtp | tcp | timer | udp }

打开 ASPF 调试开关

undo debugging aspf { all | detail | events | ftp | h323 | http | rtsp | session | smtp | tcp | timer | udp }

关闭 ASPF 调试开关

下面在防火墙上具体配置一个 ASPF 策略，来检测通过防火墙的 FTP 和 HTTP 流量。如果该报文是内部网络用户发起的 FTP 和 HTTP 连接的返回报文，则允许其通过防火墙进入内部网络，其他报文被禁止；并且，此 ASPF 策略能够过滤掉来自服务器 2.2.2.11 的 HTTP 报文中的 Java Applets。本例可以应用在本地用户需要访问远程网络服务的情况下。配置的基本步骤如下：

在 ASPF 路由器上配置允许防火墙

[Quidway] firewall enable

配置访问控制列表 111，以拒绝所有 TCP 和 UDP 流量进入内部网络，ASPF 会为允许通过的流量创建临时的访问控制列表

[Quidway] acl number 111 [Quidway-acl-adv-111] rule deny

创建 ASPF 策略，策略号为 1，该策略检测应用层的两个协议： FTP 和 HTTP 协议，并定义没有任何行为的情况下，这两个协议的超时时间为 3000s

[Quidway] aspf-policy 1

[Quidway-aspf-policy-1] detect ftp aging-time 3000

[Quidway-aspf-policy-1] detect http aging-time 3000

[Quidway-aspf-policy-1] detect http java-list 1

配置访问控制列表 1，以过滤来自站点 2.2.2.11 的 Java Applets

[Quidway] acl number 1

[Quidway-acl-basic-1] rule deny source 2.2.2.11 0

[Quidway-acl-basic-1] rule permit any

在接口上应用 ASPF 策略

[Quidway-Serial1/0/0] firewall aspf 1 outbound

在接口上应用访问控制列表 111

[Quidway-Serial1/0/0] firewall packet-filter 1 inbound

第 10 章 电子商务网站的安全

随着 Internet 的发展，电子商务已经逐渐成为人们进行商务活动的新模式。电子商务在国外已经成为一种很常见的购物方式，而在我国正处于起步阶段。电子商务是端对端的网上交易，它必须具有强有力的安全防范措施，并提供数据的完整性、保密性和不可否认性，因为网上交易使用信用卡、个人账号等私人秘密。我国电子商务正在崛起，电子商务的发展前景十分诱人，而其安全问题也变得越来越突出，如何建立一个安全、快捷的电子商务应用环境，对信息提供足够的保护，已经成为商家和用户都十分关心的话题。所以要开展电子商务，就必须充分了解电子商务中应该注意的安全问题。目前，在电子商务网站中应用最多是的客户机和服务器中的安全模式。在本章中将详细讨论电子商务中电子商务站点的安全。

10.1 电子商务的安全概述

安全体系在第 1 章中已经讲到，它包括物理系统的安全、操作系统的安全、网络安全及电子商务站点的安全，即 Web 的安全性，在前几章中分别介绍了相关的安全知识，如操作系统安全等。在这一章中将着重讲述电子商务站点的安全。

10.1.1 电子商务站点的安全准则

电子商务站点的安全准则如下：

- 信息的完整性与真实性。
- 信息的保密性。
- 信息的不可否认性（即不可抵赖性）。
- 确保消费者隐私的安全。
- 保护消费者的机密。
- 确保服务器的安全。

- 身份认证。
- 使用安全策略。
- 禁用范例和文献。
- 指定命令的安全级别。
- 备份与恢复。

10.1.2 电子商务安全体系

电子商务安全解决方案分为 4 个层次：基础设备安全、终端设备安全、网络设备安全和系统设备安全。如图 10-1 所示。

电子商务安全解决方案

基础设备（如加密卡、身份识别卡等）安全

终端设备（如传真密码机、电话密码机等）安全

安全技术（如身份证技术、访问控制技术、加密技术、智能卡技术、虚拟专用网技术及 PKI 技术等）

系统设备（如安全服务器、安全加密套件和金融加密机等）安全

网络设备（如安全路由器、防火墙及安全协议等）安全

图 10-1

一般的现代电子商务实现的解决方案如图 10-2 所示。

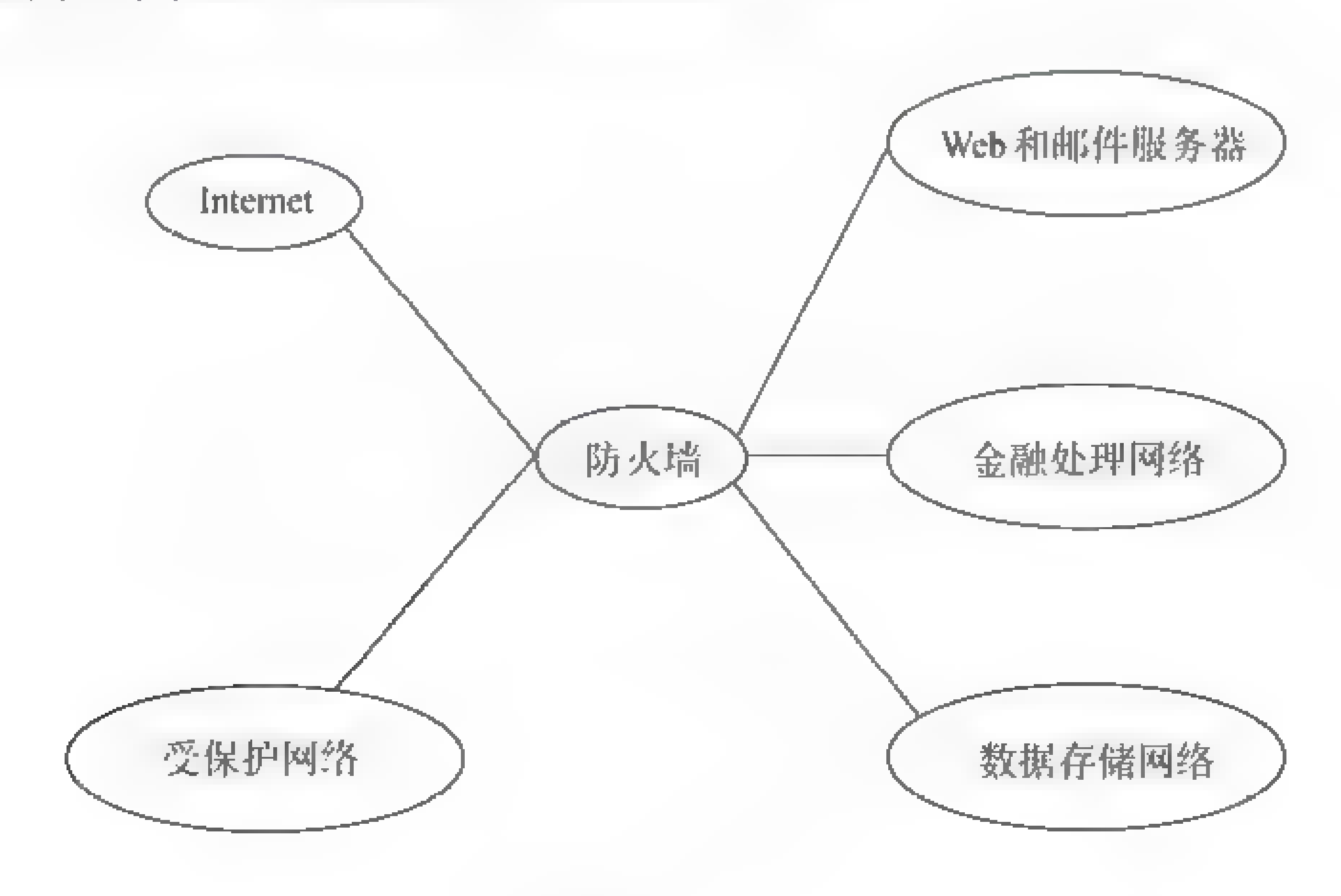


图 10-2

10.2　电子商务中所使用的安全技术

在网络安全中所包括的安全技术有密码技术、身份认证、访问控制、虚拟专用网（VPN）、公共密钥基础设施（PKI）及智能卡等。而在电子商务中用到了上述安全技术的几种，下面逐一介绍。

10.2.1　密码技术

在传统的电子商务中使用密码技术，就是使用公钥（非对称加密）和对称加密相结合的方法提供通信、保密性认证和消息完整性，客户机和服务器能通过防止窃听篡改和伪造消息的途径进行通信。公钥可以给任何请求它的应用程序或用户，私钥则只有它的所有者知道。除了公钥/私钥对，电子商务还使用数字证书（digital certificate），这是一些被 Certificate Authorities 所发布的文件，它们作为应用程序或用户的信用卡。数字证书是一些文本文件，这些文本文件包含了识别应用程序或用户身份的信息。证书中也包含了应用程序或用户的公钥。

下面介绍非对称加密（公开）和对称密钥加密。

1．对称加密算法（Symmetric Encryption）

在对称加密中，数据信息的传送，加密及接收解密都需用到共享的钥匙，也就是说加密和解密共用一把钥匙。例如，Mike 想送一张订单给 Bob，只有 Bob 可以订它。Mike 将这张订单（里面的文字）用一个加密钥匙加密之后，将这个加过密的订单（密码文字）寄给了 Bob。所谓加密就是将数据打乱，使得除了特定的收信人之外，所有的人都无法看懂它。对称加密最常用的一种方式是资料加密标准（Data Encryption Standard，DES）。所有的参与者都必须彼此了解，而且完全互相信任，因为他们每一个人都有一份钥匙的珍藏副本。如果传送者和接收者位于不同的地点，他们在面对面的会议时或是在公共传输系统（电话系统或邮局服务）时，当密钥在被互相交换时，确定不会被偷听。只要有人在钥匙传送的途中偷听到或者拦截下钥匙，他就可以用这个钥匙来读取所有加密了的数据信息。

2．非对称加密算法（Asymmetric Encryption）

在非对称加密算法中，利用了两把钥匙，一个钥匙用来将数据信息加密，而用另一把不同的钥匙来解密。这两把钥匙之间有数学关系，所以用一个钥匙加密过的资料只能用相对的另一个钥匙来解密。非对称加密异于两方都用同一个密钥的对称加密算法，公钥密码法对每一个人都使用一对钥匙，其中一个是公开的，而另一个是私密的。公共钥匙可以在互联网上明文传送，而私密钥匙则必须加以保密，只有持有人知道它的存在。但这两种钥匙都必须加以保证，防止被修改。也就是每个人都有一对密钥，一个私钥和一个公钥，他们在数字上相关，在功能上不同。一个密钥锁上的用另一个可以打开。SSL 使用公钥加密（Public Key Cryptography），该技术使用两个加密的密钥来保证会话的安全，公共钥匙加密算法主要有两种用途。

- 数据加密：发送者用接收者的公钥对要发送的数据加密，接收者用自己的私钥对接收到的数据解密，第三者由于不知道接收者的私钥而无法破译该数据。
- 身份确认：发送者可以用自己的私钥对要发送的数据做一"数字签名"，接收者通过验证"数字签名"就可准确确定数据的来源。公共钥匙加密算法又称为非对称加密算法，常见的有 RSA、DSA 等。

公共密钥方案较保密密钥方案处理速度慢，因此，通常把公共密钥与专用密钥技术结合起来实现最佳性能。即用公共密钥技术在通信双方之间传送专用密钥，而用专用密钥来对实际传输的数据加密解密。另外，公钥加密也用来对专用密钥进行加密，而 SSL 就采用了两者的结合。

10.2.2 数字签名

公钥体系中有公钥和私钥，私钥保持私有，只有拥有者才知道，公钥分布广泛（通常作为公共证书的一部分），因此，任何人都能用公钥加密数据，而只有私钥拥有者才能解密。另外，私钥拥有者用私钥加密数据，任何拥有公钥的人都能解除开，被称为数字签名。在这种情况下，签名者产生一个数字信息（例如 HASH）使用协商好的算法，然后用私钥加密。接收者能验证私钥拥有者发送的消息，用签名者的公钥解开加密的信息，并产生收到信息的相匹配的摘要。

RSA 公钥体系就可以用于对数据信息进行数字签名。所谓数字签名就是信息发送者用其私钥对从所传报文中提取出的特征数据或称数字指纹进行 RSA 算法解密运算操作，得到发信者对该数字指纹的签名函数 H(m)。签名函数 H(m)从技术上标识了发信者对该电文的数字指纹的责任。因为发信者的私钥只有他本人才有，所以他一旦完成了签名便保证了发信人无法抵赖曾发过该信息（即不可抵赖性）。经验证无误的签名电文同时也确保信息报文在经签名后未被篡改（即完整性）。当信息接收者收到报文后，就可以用发送者的公钥对数字签名的真实性进行验证。例如美国参议院已通过了立法，数字签名与手书签名的文件具有同等的法律效力。

在数字签名中有重要作用的数字指纹，是通过一类特殊的散列函数（HASH 函数）生成的，对这些 HASH 函数的特殊要求是：

- 接收的输入报文数据没有长度限制。
- 对任何输入报文数据生成固定长度的摘要（数字指纹），并输出。
- 从报文能方便地算出摘要。
- 难以对指定的摘要生成一个报文，而由该报文可以算出该指定的摘要。
- 难以生成具有相同摘要的两个不同的报文。

10.3 电子商务中的认证

公用密钥的优点就在于，也许你并不认识某一实体，但只要你的服务器认为该实体的 CA 是可靠的，就可以进行安全通信，而这正是 Web 商务所要求的，例如信用卡购物。服务方对自己的资源可根据客户 CA 的发行机构的可靠程度来授权。SSL 使用数字证书

（Digital Certificate），这是一些被CA（Certificate Authority）所发布的文件，它们作为应用程序或用户的信用卡。数字证书是一些文本文件，这些文本文件包含了识别应用程序或用户身份的信息。证书中也包含了应用程序或用户的公钥。CA 是可信赖的代理，负责确认应用程序或用户身份，并为识别身份发布数字证书。一个第三方的 CA 组织的例子是 VeriSign：www.verisign.com。

Internet 信息服务管理员可以使用微软 Certificate Service 作为自己的 CA，在自己的机构中按照需要发布或撤销数字证书。

安装在 IIS 服务器上用来为服务器提供身份证明的证书叫做服务器证书，安装在客户浏览器端的叫做客户证书。服务器和客户端的证书都必须被 CA 签发（证明是合法的）。用来标识 CA 的证书就是所谓的站点证书或 CA 证书，它是 CA 签发的。

证书会过期，如果需要，证书可以撤销，CA 保持被撤销证书的列表，该证书列表用来检查数字证书持有者的身份。

下面介绍相关的概念。

10.3.1　认证机构

CA 是证书的签发机构。构建密码服务系统的核心内容是如何实现密钥管理。公钥体制涉及一对密钥，即私钥和公钥。私钥只由持有者秘密掌握，无须在网上传送，而公钥是公开的，需要在网上传送，故公钥体制的密钥管理主要是公钥的管理问题。目前较好的解决方案是引进证书机制。

1．证书

证书是公开密钥体制的一种密钥管理媒介。它是一种权威性的电子文档，形同网络计算环境中的一种身份证，用于证明某一主体（如人、服务器等）的身份及其公开密钥的合法性。在使用公钥体制的网络环境中，必须向公钥的使用者证明公钥的真实合法性。因此，在公钥体制环境中，必须有一个可信的机构来对任何一个主体的公钥进行公证，证明主体的身份以及它与公钥的匹配关系。CA 正是这样的机构，它的职责是验证并标识证书申请者的身份；确保 CA 用于签名证书的非对称密钥的质量；确保整个签证过程的安全性，确保签名私钥的安全性；管理证书材料信息（包括公钥证书序列号、CA 标识等）；确定并检查证书的有效期限；确保证书主体标识的唯一性，防止重名；发布并维护作废证书表；对整个证书签发过程做日志记录；向申请人发通知等。

CA 也拥有一个证书（内含公钥），当然，它也有自己的私钥，所以它有签字的能力。网上的公众用户通过验证 CA 的签字从而信任 CA，任何人都应该可以得到 CA 的证书（含公钥），用以验证它所签发的证书。

如果用户想得到一份属于自己的证书，他应先向 CA 提出申请。在 CA 判明申请者的身份后，便为他分配一个公钥，并且 CA 将该公钥与申请者的身份信息绑在一起，并为之签字后，便形成证书发给那个用户（申请者）。

如果一个用户想鉴别另一个证书的真伪，他就用 CA 的公钥对那个证书上的签字进行

验证（如前所述，CA 签字实际上是经过 CA 私钥加密的信息，签字验证的过程还伴随使用 CA 公钥解密的过程），一旦验证通过，该证书就被认为是有效的。CA 除了签发证书之外，它的另一个重要作用是证书和密钥的管理。

由此可见，证书就是用户在网上的电子个人身份证，同日常生活中使用的个人身份证作用一样。CA 相当于网上公安局，专门发放、验证身份证，所以是发放和管理用户的数字证书。

由于认证机构发放的证书是供各种各样的应用程序适用的，因此它必须遵循一定的工业标准。

1）加密标准

数字证书及数字签字实质是信息加密，通用的加密标准有如下算法。

- 对称加密算法：如 DES、Tripl-DES、RC2、RC4 和 CAST 等。
- 非对称加密算法：如 RSA、DSA 和 Diffie-Hellman 等。
- 散列算法：如 SHA-1、MD5 等。

2）证书标准

通用的证书标准是 X.509。

2. 认证中心

认证中心就是一个负责发放和管理数字证书的权威机构。对于一个大型的应用环境，认证中心往往采用一种多层次的分级结构，各级的认证中心类似于各级行政机关，上级认证中心负责签发和管理下级认证中心的证书，最下一级的认证中心直接面向最终用户。处在最高层的认证中心，它是所有人公认的权威，如人民银行总行的 CA。

认证中心有颁发、更新、查询、作废和归档 5 项功能。

1）颁发功能

中心接收、验证用户（包括下级认证中心和最终用户）的数字证书的申请，将申请的内容进行备案，并根据申请的内容确定是否受理该数字证书申请。如果中心接受该数字证书申请，则进一步确定给用户颁发何种类型的证书。新证书用认证中心的私钥签名以后，发送到目录服务器供用户下载和查询。为了保证消息的完整性，返回给用户的所有应答信息都要使用认证中心的签名。

2）更新功能

认证中心可以定期更新所有用户的证书，或者根据用户的请求来更新用户的证书。

3）查询功能

证书的查询可以分为两类，一类是证书申请的查询，认证中心根据用户的查询请求返回当前用户证书申请的处理过程；另一类是用户证书的查询，这类查询由目录服务器来完成，目录服务器根据用户的请求返回适当的证书。

4）作废功能

当用户的私钥由于泄密等原因造成用户证书需要申请作废时，用户需要向认证中心提出证书作废请求，认证中心根据用户的请求确定是否将该证书作废。

另外一种证书作废的情况是证书已经过了有效期，认证中心自动将该证书作废。认证中心通过维护证书作废列表完成上述功能。

5）归档功能

证书具有一定的有效期，证书过了有效期之后就被作废，但是不能将作废的证书简单地丢弃，因为有时我们可能需要验证以前的某个交易过程中产生的数字签名，这时就需要查询作废的证书。基于此类考虑，认证中心还应当具备管理作废证书和作废私钥的功能。

安全认证是维持电子商务活动正常进行的保证，它涉及安全管理、加密处理、PKI 及认证管理等重要问题。目前已经有一套完整的技术解决方案可以应用。采用国际通用的 PKI 技术、X.509 证书标准和 X.500 信息发布标准等技术标准可以安全发放证书，进行安全认证。当然，认证机制还需要法律法规支持。安全认证需要的法律问题包括信用立法、电子签名法、电子交易法、认证管理法律等。在我国，安全认证工作已经应用到电子商务工程中，相信这将大大促进我国电子商务的开展。

10.3.2 数字证书

数字证书是一种能在完全开放系统使用的。如互联网络的用户群（绝不是几个人互相信任的小集体），在这个用户群中，从法律角度讲，用户彼此之间都不能轻易信任。所以公钥加密体系采取了另一个办法，将公钥和公钥的主人名字联系在一起，再请一个大家都信得过的有信誉的公正、权威机构确认，并加上这个权威机构的签名，就形成了证书。

由于证书上有权威机构的签字，所以大家都认为证书上的内容是可信任的；又由于证书上有主人的名字等身份信息，别人就很容易地知道公钥的主人是谁。公钥体制涉及一对密钥，即私钥和公钥。私钥只由持有者秘密掌握，无须在网上传送，而公钥是公开的，需要在网上传送，故公钥体制的密钥管理主要是公钥的管理问题。目前较好的解决方案是引进证书机制。

1. 证书的格式与证书的发放

证书是公开密钥体制的一种密钥管理媒介。它是一种权威性的电子文档，形同网络计算环境中的一种身份证，用于证明某一主体（如人、服务器等）的身份及其公开密钥的合法性。在使用公钥体制的网络环境中，必须向公钥的使用者证明公钥的真实合法性。因此，在公钥体制环境中，必须有一个可信的机构来对任何一个主体的公钥进行公证，证明主体的身份以及它与公钥的匹配关系。数字证书格式的通用标准是 X.509。根据该标准，数字证书应有下列信息：

- 版本号。
- 序列号。
- 签字算法。
- 发出该证书的认证机构。
- 有效期限。
- 主题信息，包括持有人的姓名、服务处所等信息。
- 公共密钥信息。
- 认证机构的数字签字。

在 X.509 标准的扩展部分，认证机构可以说明该证书的附加信息，如密钥用途等。

用户想获得认证机构的证书，首先要向认证机构提出申请，说明自己的身份。认证机构在认真查验用户的身份后，向用户发出相应的数字证书。

认证机构在发放证书时要遵循一定的准则，如要保证自己发出的证书的序列号没有相同的，没有两个不同的实体获得的证书中的主题内容是一致的，不同主题内容的证书所包含的公开密钥要不相同等。

2. 证书的管理

认证机构应有一个证书管理机构来管理它发出的所有证书，包括如下管理功能。

- 用户能方便地查找各种证书及已经撤销的证书。用户在验证发送方数字签字时需要验证用户的身份，这就要检验发送方的数字证书。由于该证书可能在其有效期限内被认证机构撤销，用户往往要检查认证机构的已撤销证书。因此，能否给用户提供方便的证书查询功能，是认证机构是否成功的重要标准之一。
- 根据用户请求或其他相关信息撤销用户的证书。用户的身份并不是一成不变的。如用户的服务处所改变后，用户的身份也就改变了。这时，原来的证书虽在有效期限内，但已无意义，认证机构就应该根据用户的请求，撤销该证书。
- 根据证书的有效期限自动地撤销证书。
- 完成证书数据库的备份工作。

3. 证书库

证书库是证书的集中存放地，它与网上"白页"类似，是 Internet 上的一种公共信息库，用户可从此处获得其他用户的证书和公钥。构造证书库的最佳方法是采用支持 LDAP 协议的目录系统，用户或相关的应用通过 LDAP 来访问证书库。系统必须确保证书库的完整性，防止伪造、篡改证书。

4. 密钥备份及恢复系统

如果用户丢失了用于解密数据的密钥，则密文数据将无法被解密，从而造成数据丢失。为避免这种情况的出现，PKI 应该提供备份与恢复解密密钥的机制。密钥的备份与恢复应该由可信的机构来完成，例如 CA 可以充当这一角色。值得强调的是，密钥备份与恢复只能针对解密密钥，签名私钥不能够做备份。

5. 证书作废处理系统

证书作废处理系统是 PKI 的一个重要组件。同日常生活中的各种证件一样，证书在 CA 为其签署的有效期以内也可能需要作废。例如，A 公司的职员 a 辞职离开公司，这就需要终止 a 证书的生命期。为实现这一点，PKI 必须提供作废证书的一系列机制。作废证书有三种策略：一是作废一个或多个主体的证书；二是作废由某一对密钥签发的所有证书；

三是作废由某CA签发的所有证书。

作废证书一般通过将证书列入作废证书表（CRL）来完成。通常，系统中由CA负责创建并维护一张及时更新的CRL，而由用户在验证证书时负责检查该证书是否在CRL之列。CRL一般存放在目录系统中。证书的作废处理必须在安全及可验证的情况下进行，系统还必须保证CRL的完整性。

6. 客户端证书处理系统

客户端证书与浏览器有关，证书申请人可以通过浏览器申请和下载证书，并安装在浏览器上使用。

7. 认证算法

认证算法采用X.509电子证书标准，通过使用RSA算法进行数字签名来实现。

1）服务器的认证

在加密算法和会话密钥产生的两对密钥中，服务器方的写密钥和客户方的读密钥、客户方的写密钥和服务器方的读密钥分别是一对私有、公有密钥。对服务器进行认证时，只有用正确的服务器方的写密钥加密CLIENT-HELLO消息形成的数字签名，才能被客户正确地解密，从而验证服务器的身份。若通信双方不需要新的密钥，则它们各自所拥有的密钥已经符合上述条件；若通信双方需要新的密钥，首先服务器方在SERVER-HELLO消息中的服务器证书中提供服务器的公有密钥，服务器用其私有密钥才能正确地解密由客户方使用服务器的公有密钥加密的MASTER-KEY，从而获得服务器方的读密钥和写密钥。

2）客户的认证

同样，只有用正确的客户方的写密钥加密的内容，才能被服务器方用其读密钥正确地解开。当客户收到服务器方发出的REQUEST-CERTIFICATE消息时，客户首先使用MD5消息散列函数获得服务器方信息的摘要，服务器方的信息包括KEY-MATERIAL-0、KEY-MATERIAL-1、KEY-MATERIAL-2、CERTIFICATE-CHALLENAGE-DATA（来自于REQUEST-CERTIFICATE消息）和服务器所赋予的证书（来自于SERVER-HELLO）消息。其中KEY-MATERIAL-1和KEY-MATERIAL-2是可选的，与具体的加密算法有关。然后客户使用自己的读密钥加密摘要形成数字签名，从而被服务器认证。

8. 钥匙交换（Key Exchange）

在实际应用中通常综合使用对称算法和非对称算法相结合的算法，源数据采用对称算法加密，对称算法的钥匙称为Session Key，则用非对称算法加密后在通信双方间交换，如图10-3所示。

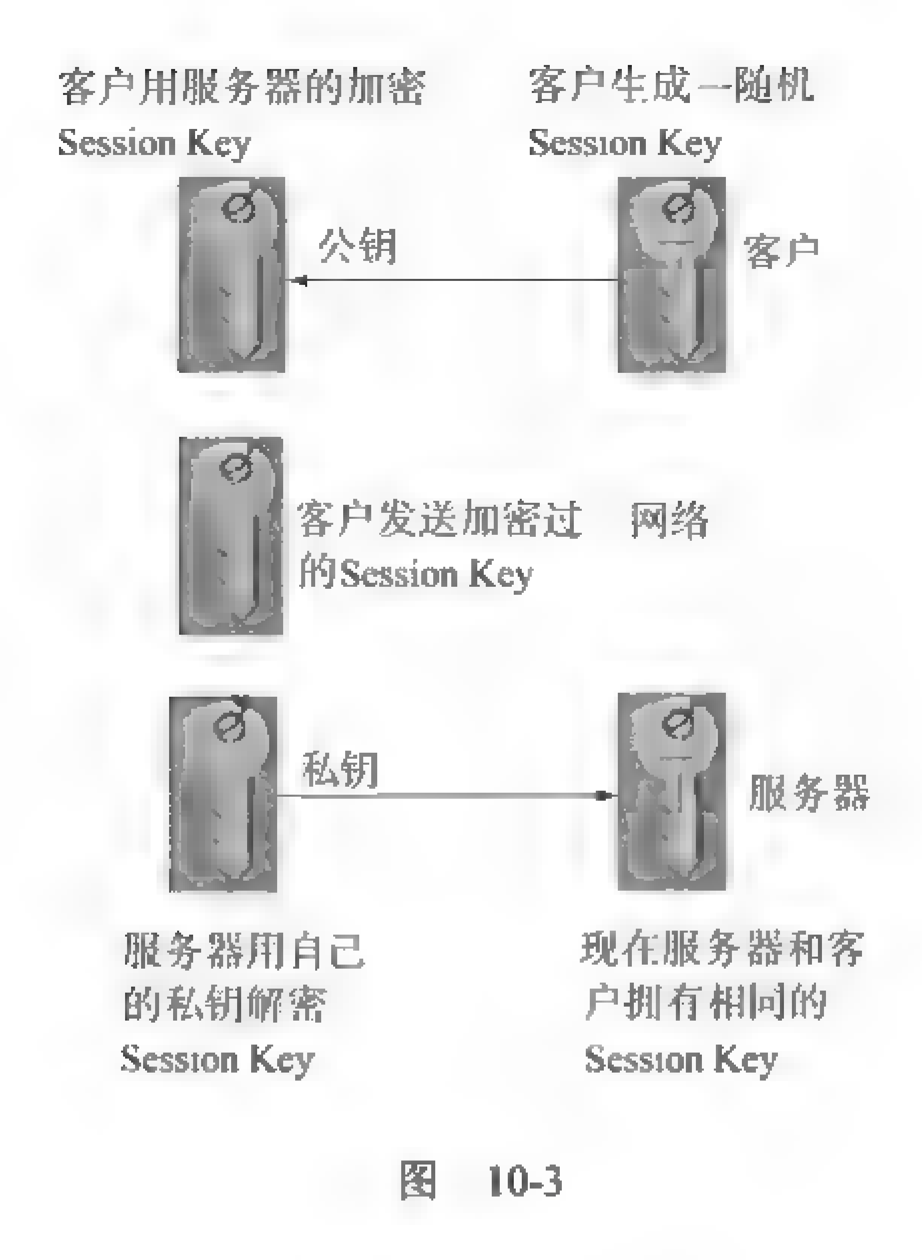


图　10-3

10.4　SSL 协议

随着计算机网络技术向整个社会各层次延伸，整个社会表现为对 Internet、Intranet 和 Extranet 等使用的更大的依赖性。随着企业间信息交互的不断增加，任何一种网络应用和增值服务的使用程度将取决于所使用网络的信息安全有无保障，网络安全已成为现代计算机网络应用的最大障碍，也是急需解决的难题之一。由于 Web 上有时要传输重要或敏感的数据，因此 Netscape（网景）公司在推出 Web 浏览器的同时，提出了安全套接（层）SSL（Secure Sockets Lager）协议，目前已有 2.0 和 3.0 版本。它包括服务器认证、客户认证（可选）、SSL 链路上的数据完整性和 SSL 链路上的数据保密性。对于电子商务应用来说，使用 SSL 可保证信息的真实性、完整性和保密性。但由于 SSL 不对应用层的消息进行数字签名，因此不能提供交易的不可否认性，这是 SSL 在电子商务中使用的最大不足。有鉴于此，网景公司在从 Communicator 4.04 版开始的所有浏览器中，引入了一种称做"表单签名(Form Signing)"的功能，在电子商务中，可利用这一功能来对包含购买者的订购信息和付款指令的表单进行数字签名，从而保证交易信息的不可否认性。SSL 采用公开密钥技术（在上节中已经详细叙述）。其目标是保证两个应用间通信的保密性和可靠性，可在服务器和客户机两端同时实现支持。SSL 协议是使用公开密钥体制和 X.509 数字证书技术保护信息传输的机密性和完整性，主要适用于点对点之间的信息传输，常用 Web Server 方式。目前，利用公开密钥技术的 SSL 协议，并已成为 Internet 上保密通信的工业标准。现行 Web 浏览器普遍将 HT-TP 和 SSL 相结合，从而实现安全通信。

目前几乎所有处理具有敏感度的资料、财务资料或者要求身份认证的网站都会使用

SSL 加密技术，SSL 可以是一种浏览器和网络服务器之间的受密码保护的安全通道。在 Internet 上访问某些网站时你会注意到在浏览器窗口的下方会显示一个锁状的小图标。这个小锁表示该网页被 SSL 保护着。SSL 是一种用于网站安全连接的协议（或技术），所谓的安全连接有两个作用，首先是 SSL 可以提供信息交互双方认证对方的身份标识。显而易见，这对与对方交换机密信息前确切了解对方身份是非常重要的。SSL 通过数字证书的技术实现，使得这一需求得以满足。另一个是它能够使数据以不可读的格式传输，以利于在不可信网络（例如 Internet）上的安全传输需要。这种不可读格式通常由加密技术实现。

10.4.1 协议概述

SSL 协议基本的目标是在两个通信应用中提供秘密的可靠的通信。这个协议由两层组成，最低层是在一些可信任的传输层之上（如 TCP/IP 记录协议）。SSL 记录协议是用于封装一系列较高层协议。一个就是封闭这样的协议如 SSL 握手协议，在应用层传输或者接收第一字节数据之前，允许服务器和客户彼此鉴定并且协商密码运算法则和加密密钥。SSL 的一个优势在于它是一个独立的应用协议。一个较高层的协议可以透明地放到 SSL 协议之上。

SSL 协议是在 Internet 基础上提供的一种保证私密性的安全协议。它能使客户与服务器应用之间的通信不被攻击者窃听，并且始终对服务器进行认证，还可选择对客户进行认证。SSL 协议要求建立在可靠的传输层协议（例如 TCP）之上。SSL 协议的优势在于它是与应用层协议独立无关的。高层的应用层协议（例如 HTTP、FTP 和 TELNET 等）能透明地建立于 SSL 协议之上。SSL 协议在应用层协议通信之前就已经完成加密算法、通信密钥的协商以及服务器认证工作。在此之后，应用层协议所传送的数据都会被加密，从而保证通信的私密性。

SSL 是一个介于 HTTP 协议与 TCP/IP 之间的可选层，SSL 在参数模型中的位置如图 10-4 所示。

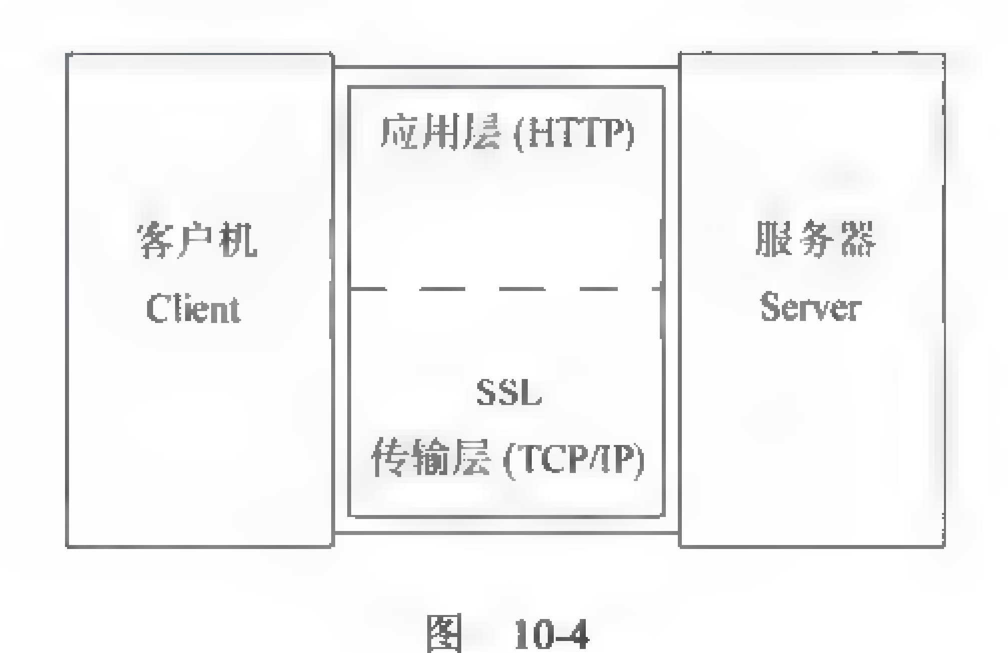


图 10-4

1. TCP/IP 层

TCP/IP 层是处理从源端到目的网络数据包的传送。TCP/IP 会话的基础是一个层（Client）和另一个对等层（Server）建立网络连接。连接用于两个对等层的会话期间，当会话结束，连接被释放。

2. 应用层

应用层定义公共的共享协议，这些协议被用于建立TCP/IP连接的通信中。例如，HTTP协议是Web客户和Web服务器用于通信的协议。应用层的会话在对一个服务器建立TCP/IP时进行初始化，服务器应使常用端口，如HTTP的典型的端口是80，所以将在服务器的80端口建立HTTP会话的TCP/IP连接。

3. SSL层

SSL层用于验证最终安全的应用层通信内容。SSL安全连接在建立对等层的鉴别和以安全的方式建立加密方式和密钥时，应用层的通信才能进行，所有输入通信量被中间的SSL层解码并且传送给应用层，在发送以前输出通信量被SSL层加密。

SSL层借助下层协议的信道安全地协商出一份加密密钥，并用此密钥来加密HTTP请求。

实际上，SSL安全服务器是典型的运行在不同的上层应用的端口，如HTTPS运行在443端口。开始使用SSL时，必须进入运行HTTPS而不是HTTPS的URL。相应地，SSL使用的默认端口为443而不是80。首先由浏览器向服务器端发送使用SSL的联络信号，其中包括加密方法和压缩模式。如果服务器支持SSL应用，就回应一个信号，也向客户发送自己的加密方法和压缩模式，包括它的证书及随机产生的数据，后者将会在执行协议时用到。

客户端用服务器发送的信息验证服务器身份，检查所连服务器名与其证书是否相符，并确定其证书是否有效。

该认证过程还包括检验证书上的数字签名，这种数字签名必须是由客户端信任的认证机构签发的公钥中的一个，它们或是预先存放在浏览器里，或是后来提供给用户的。如果发现证书与服务器名不符合，或者不是由某个认证机构签发的，客户可立即中止连接。

客户完成认证之后，就发送他自身的随机数据，并且根据所选择的密钥交换协议，来决定发送一个加密密钥还是发送一组确定该密钥的数据。这个密钥将用来产生会话密钥，以及在交换数据信息时为数据加密，同时还将为这些数据信息进行数字化签名，以维护被交换数据的完整性并对其提供认证。服务器端也可以要求客户端提供证书，并对浏览器用户进行认证。

如果双方握手成功，浏览器就向Web服务器发送完成信号，服务器也以自己的完成信号作为响应，于是双方交换加密信息和经数字签名的数据。

密钥交换在计算机操作中需要花费一段较长的时间。因此，SSL协议中还包括减少不必要的密钥交换量。假如Web浏览器连接另一个同样使用SSL的服务器，就可以向它发送一个会话标识符，如果服务器接受该标识符，那么它们就可以使用原来约定的加密、压缩算法及公用的密钥，相互间进行信息交换。

10.4.2　SSL协议连接安全的特征

SSL协议提供连接安全有三个基本的特征。

1. 私密性

这种连接是私有的，加密被用于在初始化握手且定义一个密钥，对称加密被用于数据加密（如DEC、RC4等）。

2. 确认性

对等的身份使用非对称加密或公钥加密能被鉴别（如RSA、DSS等）。

3. 可靠性

连接是可靠的，信息传送包括信息完整性检验，它使用信息验证控制（Message Authenticate Control，MAC）规则，安全信号功能（如SHA、MD5等）被用于MAC计算。

10.4.3 协议规范

SSL是一个层的协议，在每一层，信息可以包括长度、描述和内容的范围。SSL携带的信息在发送方可以随意划分数据块的大小，压缩数据，应用MAC运算法则以及加密等操作进行传送，在接收端对接收的数据进行解密、验证、解压缩，重组后，传送高层客户端。

1. 连接状态（Session and Connection State）

一个SSL会话是有状态的，SSL握手协议的职责就是调整客户和服务器的状态，因此允许每一个协议状态的机器一贯地运转，不管客户机和服务器的状态是否一致，逻辑上状态表现两次，一次被看作是目前的运转状态，并且（在握手协议过程中）再一次被看作未决的状态。另外，独立的读和写状态被维持。当客户或服务器收到了改变密码说明的信息时，它复制未决的读和写状态。当客户或服务器发送改变密码说明的信息时，它复制未决的写状态进入当前的写状态。当握手商议已完成，客户和服务器交换改变密码说明信息，然后用新达成的密码说明进行通信。一个SSL会话可以包括多重安全连接。另外，用户可以同时参与多重会话。

1）会话状态包括的元素

- 会话检验（Session Identifier）：一个任意的字节序列被服务器选取去识别一个活动的或可恢复的会话状态。
- 对等的证书（Peer Certificate）：X509.v3 [X509]对等证书，这个元素的状态可以为空。
- 压缩方法（Compression Method）：运算法则被用作先于加密技术而压缩数据。
- 密码规则（说明）（Cipher Spec）：详细说明加密运算法则（如null、des等）和一个MAC运算法则（如MD5、SHA）的数据大小。它还详细说明用密码写的特征，诸如无用信号的大小。
- 主要的密钥（Master Secret）：48字节的秘密共享在客户和服务器之间。
- 可恢复（resumable）：一个标记表明是否这个会话可被用作初始化一个新的连接。

2）连接状态包括的元素

- 随机服务器和客户（Server and Client Random）：对每一次连接，字节序列能被服务器和客户随机选取。
- 服务器写 MAC 密码（Server Write MAC Secret）：在 MAC 数据操作中使用的密钥由服务器写出。
- 客户写 MAC 秘密码（Client Write MAC Secret）：由客户写出密钥。
- 服务器写密钥（Server Write Key）：由服务器解密并由客户加密的为数据写的大量密钥。
- 客户写密钥(Client Write Key)：由客户机解密并由服务加密的为数据写的大量密钥。

2. SSL 协议栈

SSL 协议主要由 SSL 记录协议(Record Protocol)和 SSL 握手协议(Handshake Protocol)两部分组成。其中握手协议是用来协商密钥的，协议的大部分内容就是通信双方如何利用它来安全地协商出一份密钥。记录协议则定义了传输的格式。除了这两个主要协议以外，还有如 SSL 修改密文协议、SSL 警告协议等相关的协议。SSL 协议栈如图 10-5 所示。

SSL 握手协议	SSL 修改密文协议	SSL 警告协议	HTTP 传输协议
SSL 记录协议			
TCP			
IP			

图　10-5

10.5　建立安全的 Web 站点

IIS（Internet Information Server）作为当今流行的 Web 服务器之一，提供了强大的 Internet 和 Intranet 服务功能，如何加强 IIS 的安全机制，建立一个高安全性能的 Web 服务器，已成为 IIS 设置中不可忽视的重要组成部分。在本节中，将使用 SSL 把 IIS 的 Web 站点建成为一个安全站点。

IIS 的身份认证有一种安全性更高的认证，通过 SSL 安全机制使用数字证书。SSL 位于 HTTP 层和 TCP 层之间，用于建立用户与服务器之间的加密通信，确保所传递信息的安全性。SSL 是工作在公共密钥和私人密钥基础上的，任何用户都可以获得公共密钥来加密数据，但解密数据必须要通过相应的私人密钥。使用 SSL 安全机制时，首先客户端与服务器建立连接，服务器把它的数字证书与公共密钥一并发送给客户端，客户端随机生成会话密钥，用从服务器得到的公共密钥对会话密钥进行加密，并把会话密钥在网络上传递给服务器，而会话密钥只有在服务器端用私密钥才能解密，这样，客户端和服务器端建立了一个唯一的安全通道。

10.5.1　建立安全的 Web 站点应具备的条件

建立安全的 Web 站点，一般应具备以下三个条件。

1. 一台服务器

一台装有 Windows 2000 Server 的服务器。

2. 安装了 IIS 服务

在安装了 Windows 2000 Server 时一般已经安装了 IIS 服务。如果没有安装，请按照下述步骤安装。

（1）选择“开始”→“设置”→“控制面板”→“添加/删除程序”→“添加/删除 Windows 组件”命令，打开图 10-6 所示的 Windows 组件向导对话框。

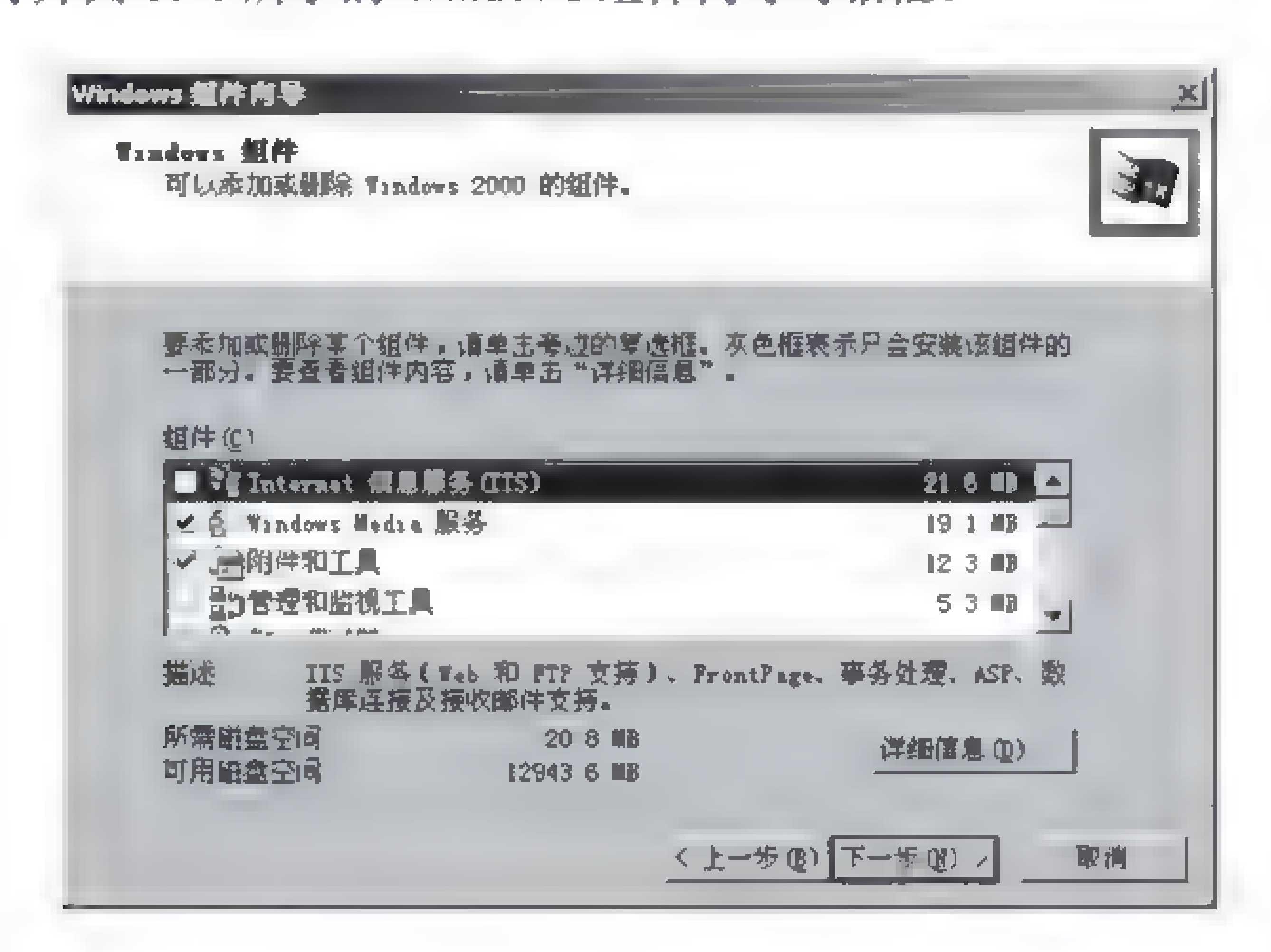


图 10-6

（2）选中“Internet 信息服务”复选框，如图 10-6 所示。

（3）单击“下一步”按钮，将出现向导中的“正在配置组件”对话框，如图 10-7 所示。

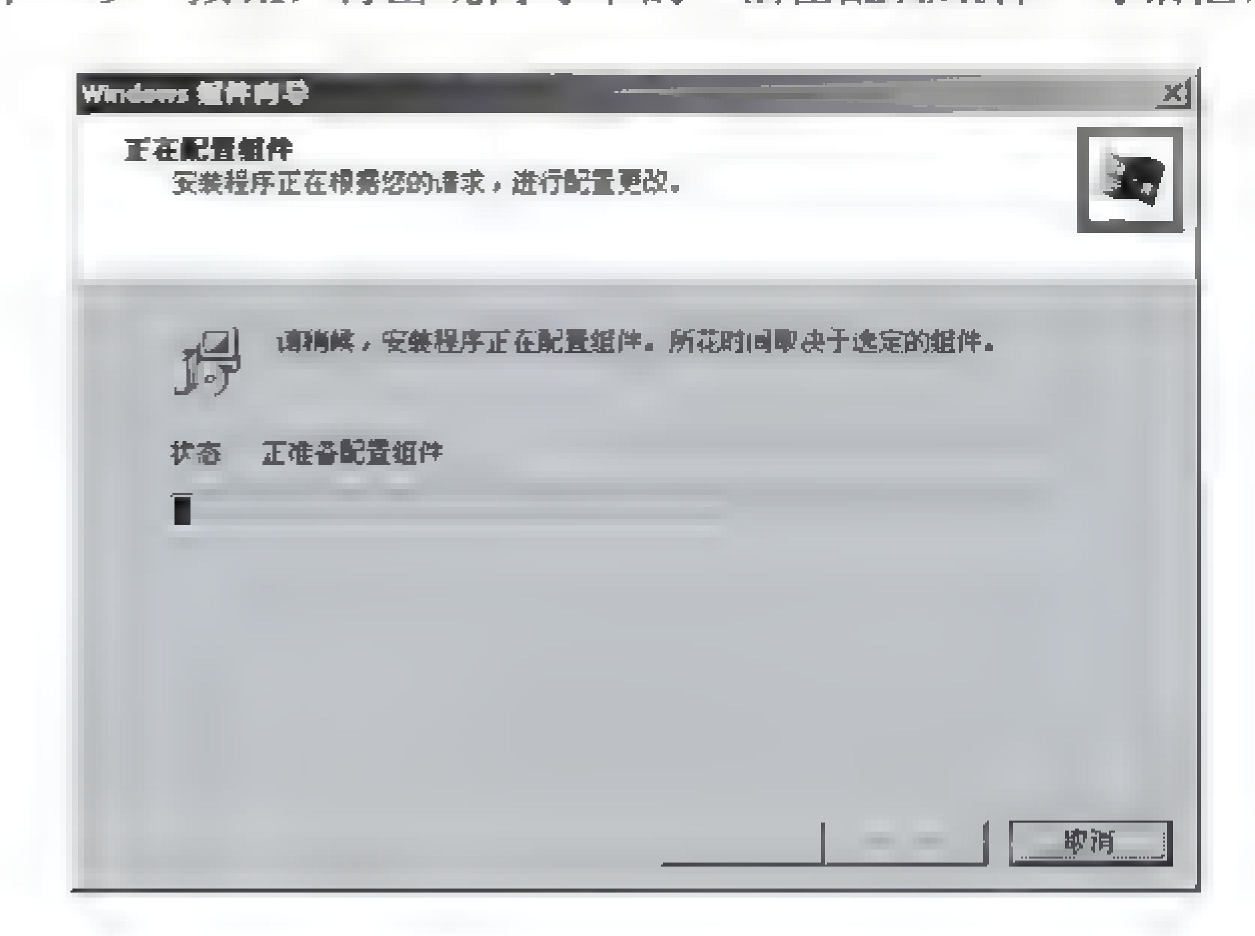


图 10-7

（4）因为复制 Windows 2000 安装盘上的相关文件，所以在相应光驱中放入安装盘，在“文件复制来源”中输入文件路径或通过“浏览”按钮找到文件复制来源的路径，如图 10-8 所示。

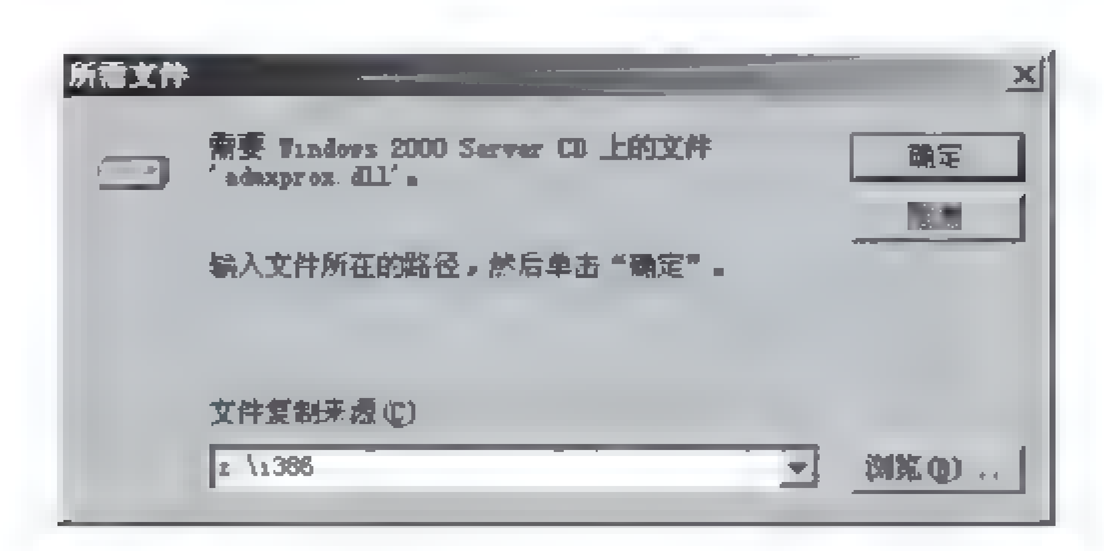


图　10-8

（5）单击“确定”按钮，开始安装 Internet 信息服务，如图 10-9 所示。

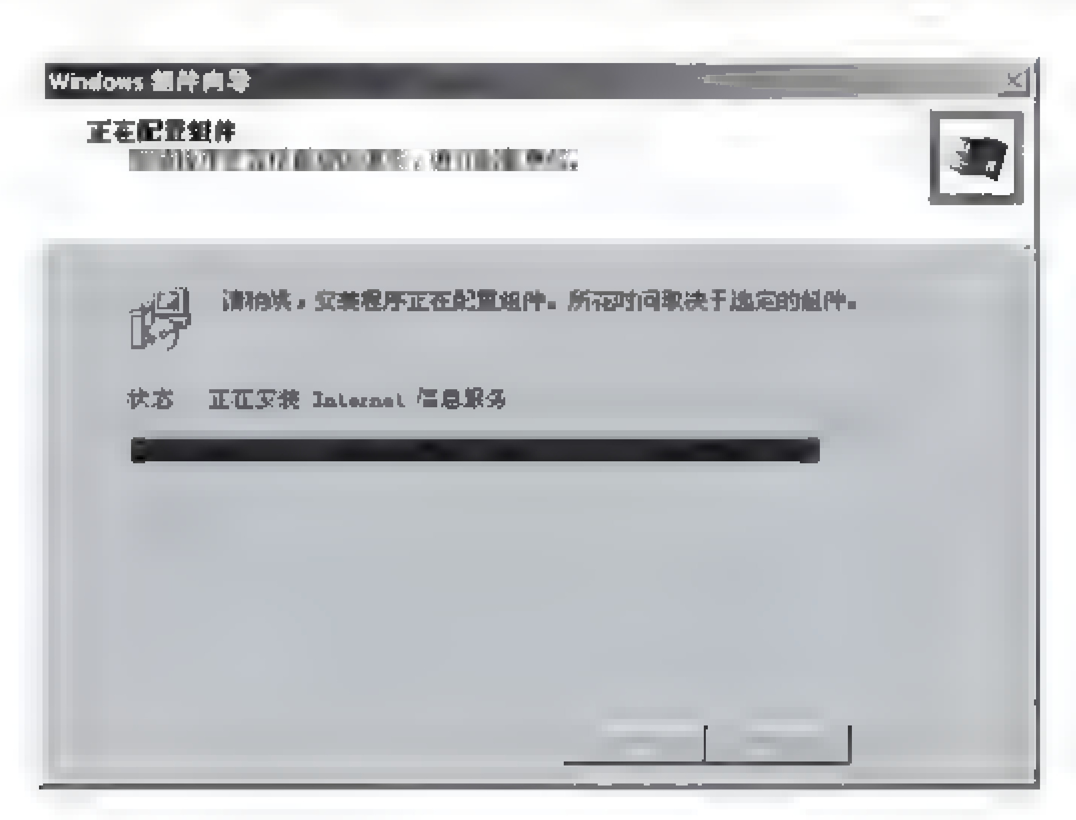


图　10-9

（6）出现“完成‘Windows 组件向导’”对话框，如图 10-10 所示，单击“完成”按钮则完成 IIS 的安装。

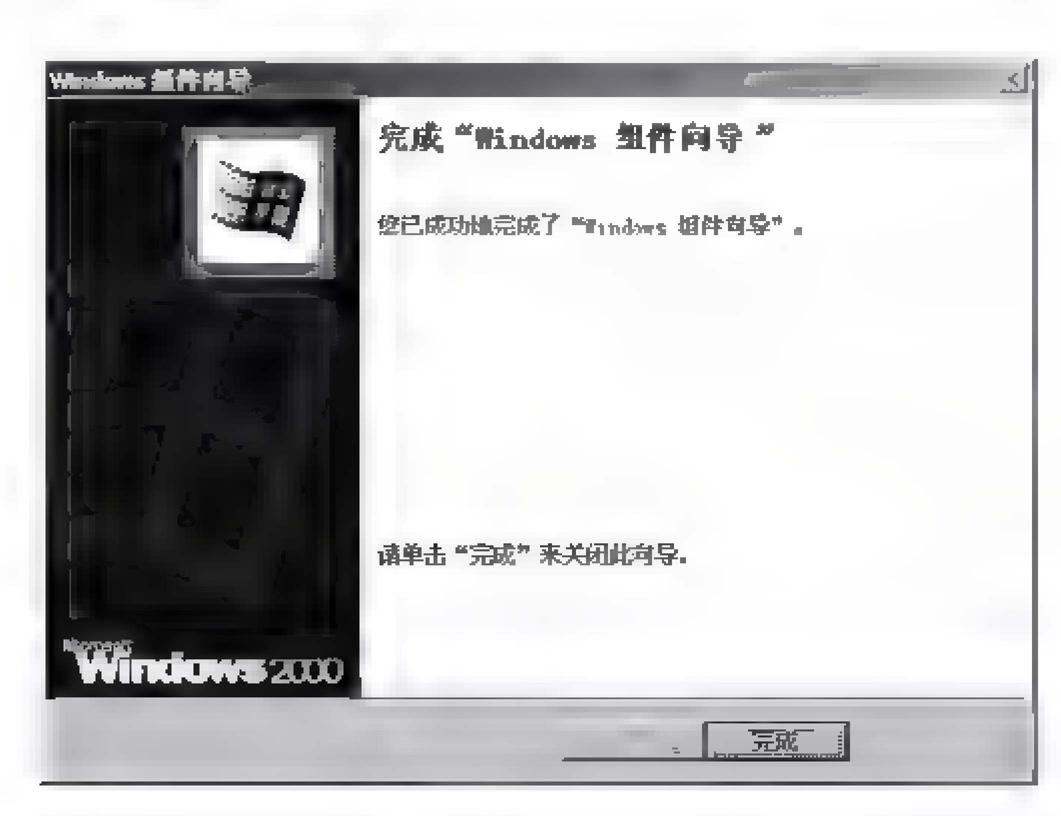


图　10-10

3. 安装证书服务

在安装证书服务之前，建立一个共享文件夹，用来保存CA证书和各种配置文件。建立共享文件夹为D:\Certificate。下面是安装证书服务的具体操作步骤。

（1）选择“开始”→“设置”→“控制面板”→“添加/删除程序”→“添加/删除Windows组件”命令，出现图10-11所示“Windows组件”对话框。

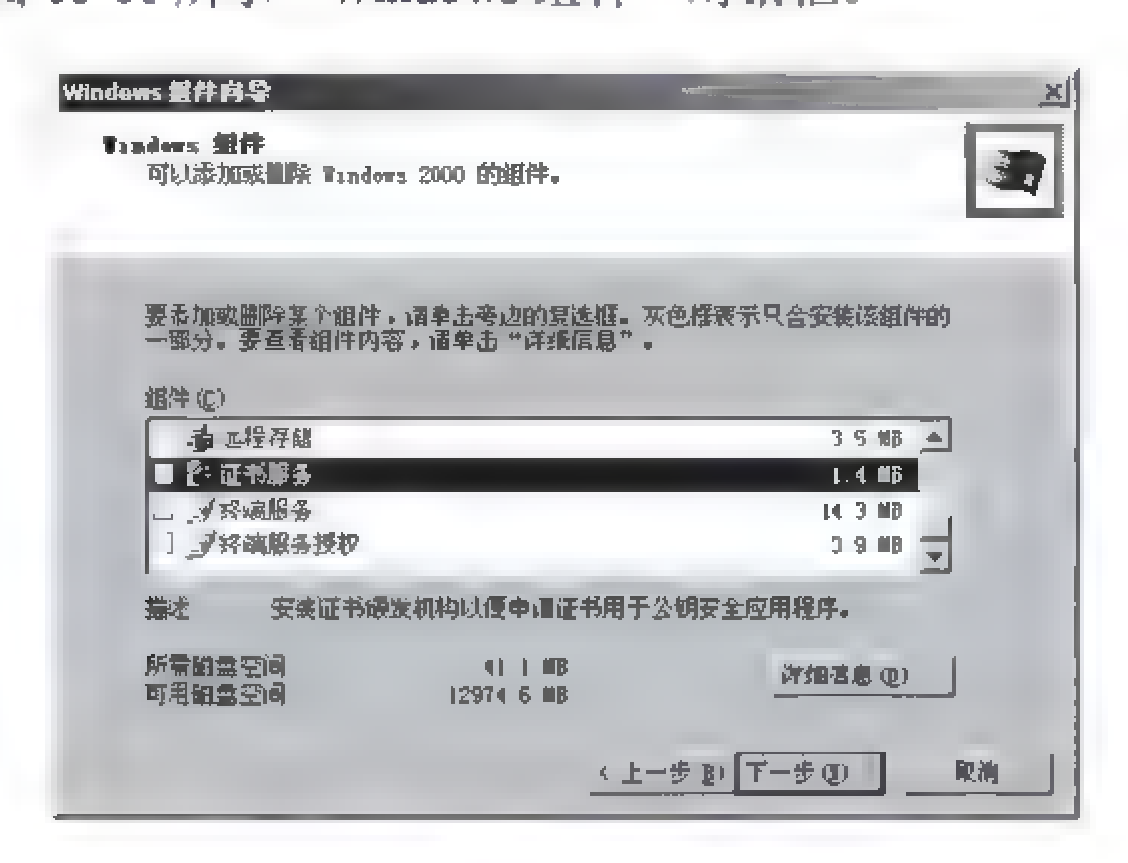


图 10-11

（2）选中“证书服务”组件的复选框，将出现图10-12所示的提示对话框。



图 10-12

（3）单击“是”按钮，出现图10-13所示的“证书颁发机构类型”对话框，其中各选项说明如下。

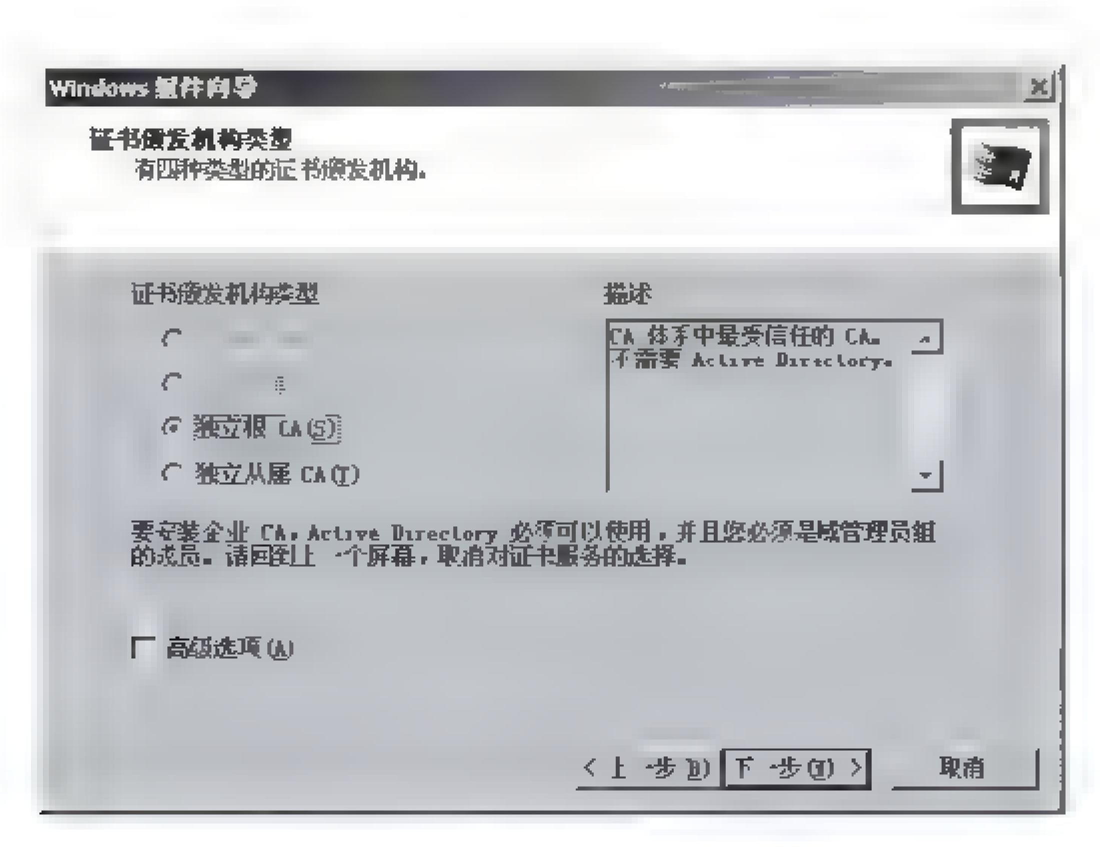


图 10-13

- 企业根 CA：与活动目录集成以判断请求者身份，与内联网一起使用。
- 企业从属 CA：属于已经存在的 Root 证书的附属证书。
- 独立根 CA：默认值，此请求将发送到一个等待队列中去，然后发送给管理员。此项不需要活动目录或第三方目录服务。
- 独立从属 CA：属于已经存在的 Root 证书。

（4）单击“下一步”按钮，出现图 10-14 所示的“CA 标识信息”对话框，在对话框中按要求进行设置，因设置较简单，这里不详细描述。

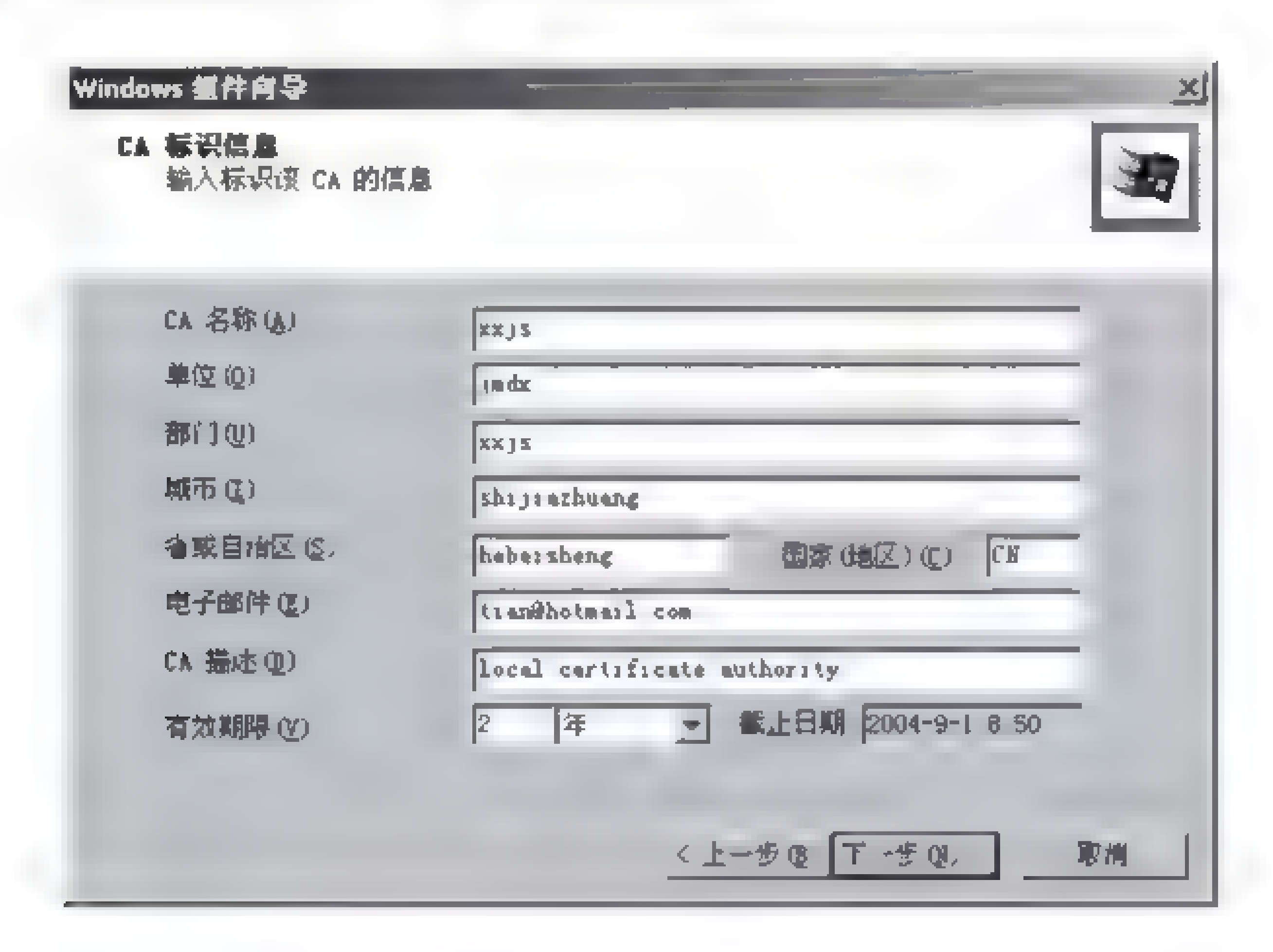


图　10-14

（5）单击“下一步”按钮，出现图 10-15 所示的“数据存储位置”对话框。

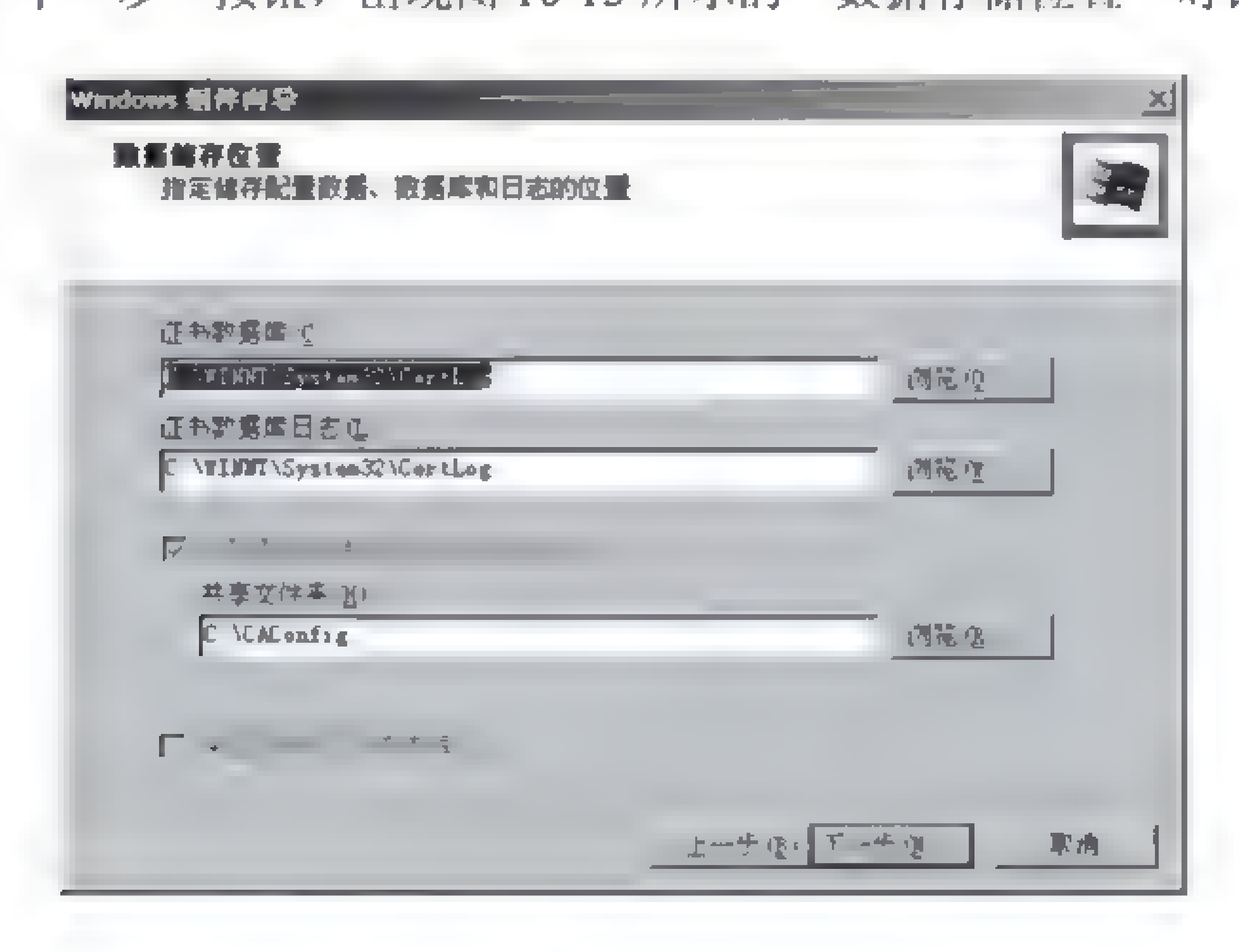


图　10-15

（6）单击“下一步”按钮，开始安装组件，出现“正在配置组件”对话框，可以看到配置进度，如图 10-16 所示。

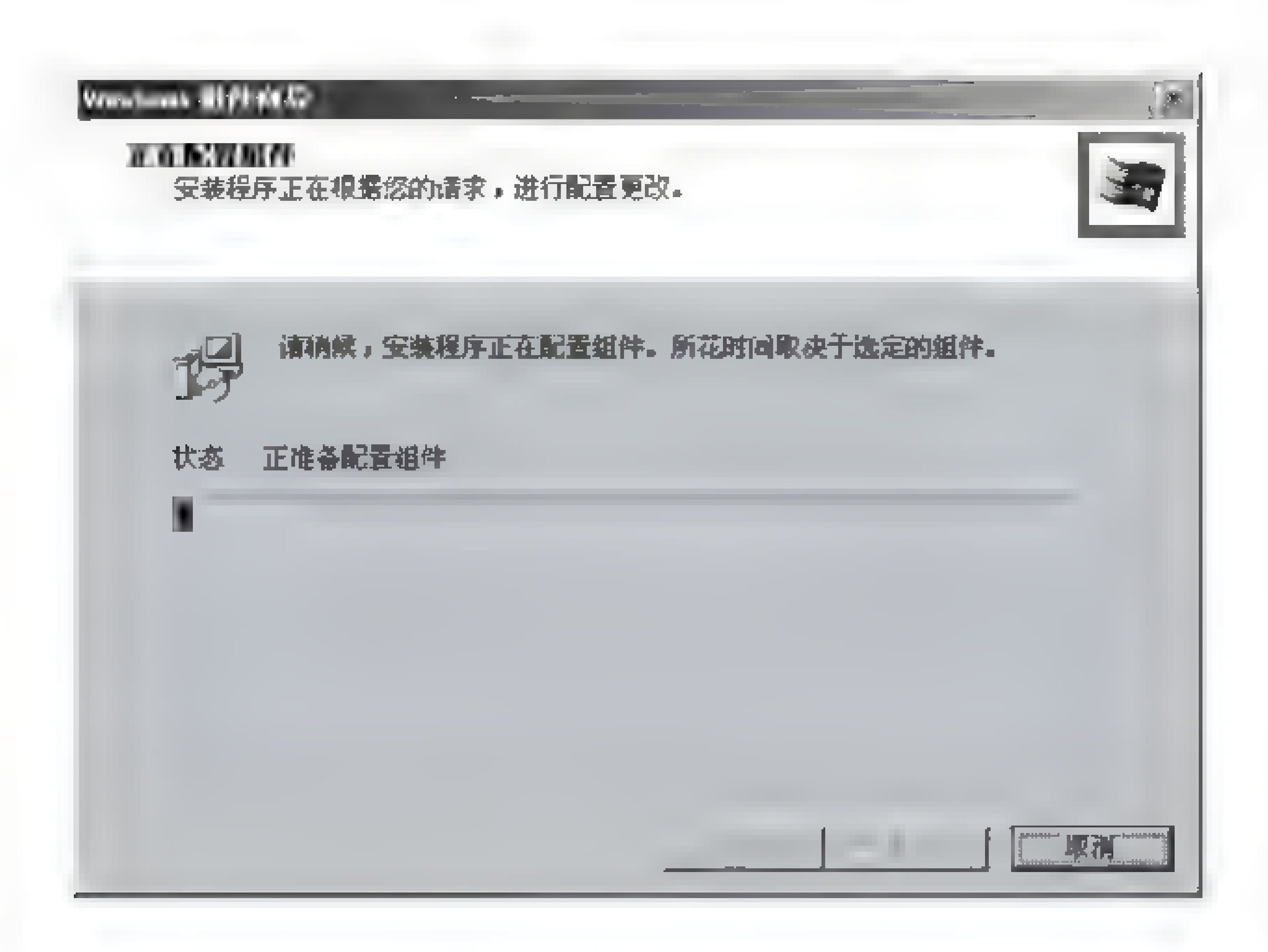


图 10-16

（7）配置完成后，单击“下一步”按钮，出现图 10-17 所示的“完成‘Windows 组件向导’”对话框，最后单击“完成”按钮，至此，Windows 组件安装完成。



图 10-17

10.5.2 建立并安装一个站点证书

建立并安装一个站点证书需要以下 7 个步骤：建立虚拟目录，建立密钥对和证书请求，向证书授权机构提交证书请求文件，证书服务器工具，安装服务器证书，在虚拟服务器上允许使用 SSL 和向客户浏览器中增加 CA 证书。

1. 建立虚拟目录

在此虚拟目录下放 Web 站点的内容，本操作以 D:\tian 为虚拟目录。

（1）首先打开 Internet 信息服务，选择“开始”→“管理工具”→“Internet 信息服务”命令，打开图 10-18 所示的“Internet 信息服务”对话框。



图　10-18

（2）双击 xeon-1g 选项，打开 IIS 服务，如图 10-19 所示。

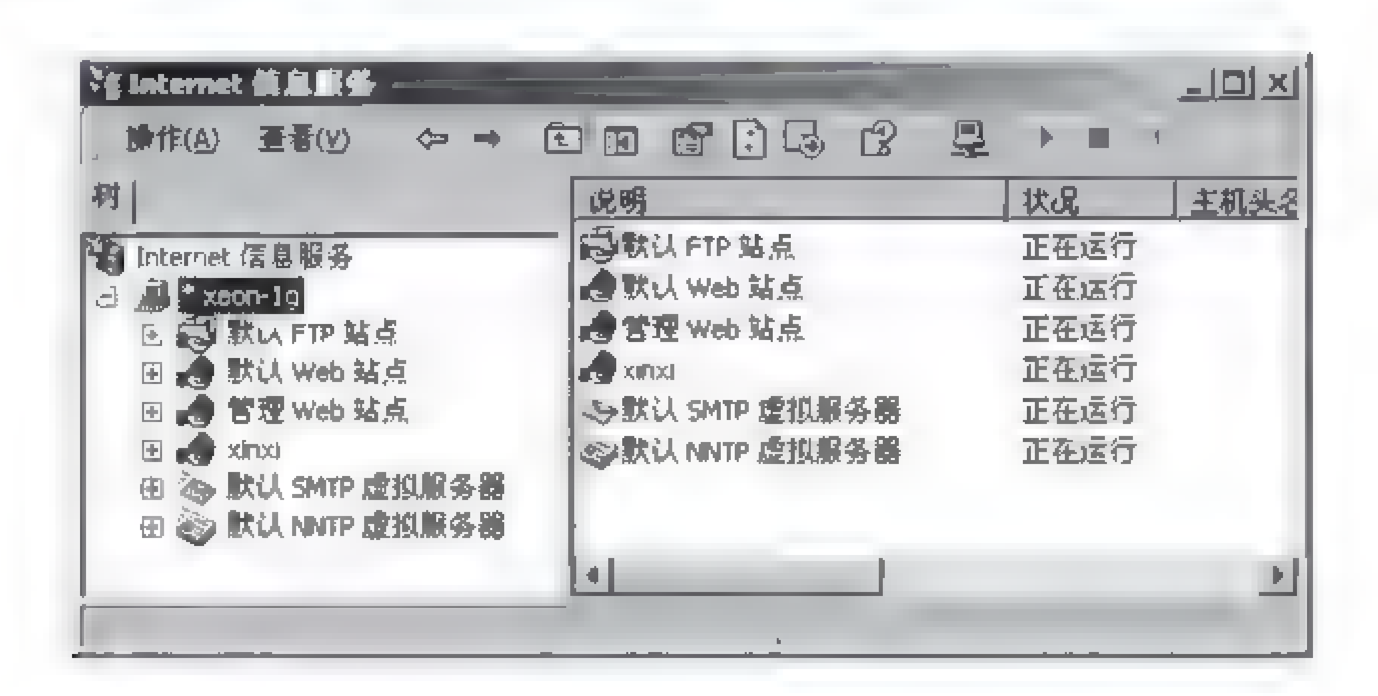


图　10-19

（3）在“默认 Web 站点”处右击，选择“新建”选项，可以建立自己的 Web 站点或虚拟目录，这里建立虚拟目录，然后将自己 Web 站点的内容放在新建的虚拟的主目录下，如图 10-20 所示。出现“虚拟目录创建向导”对话框，如图 10-21 所示。

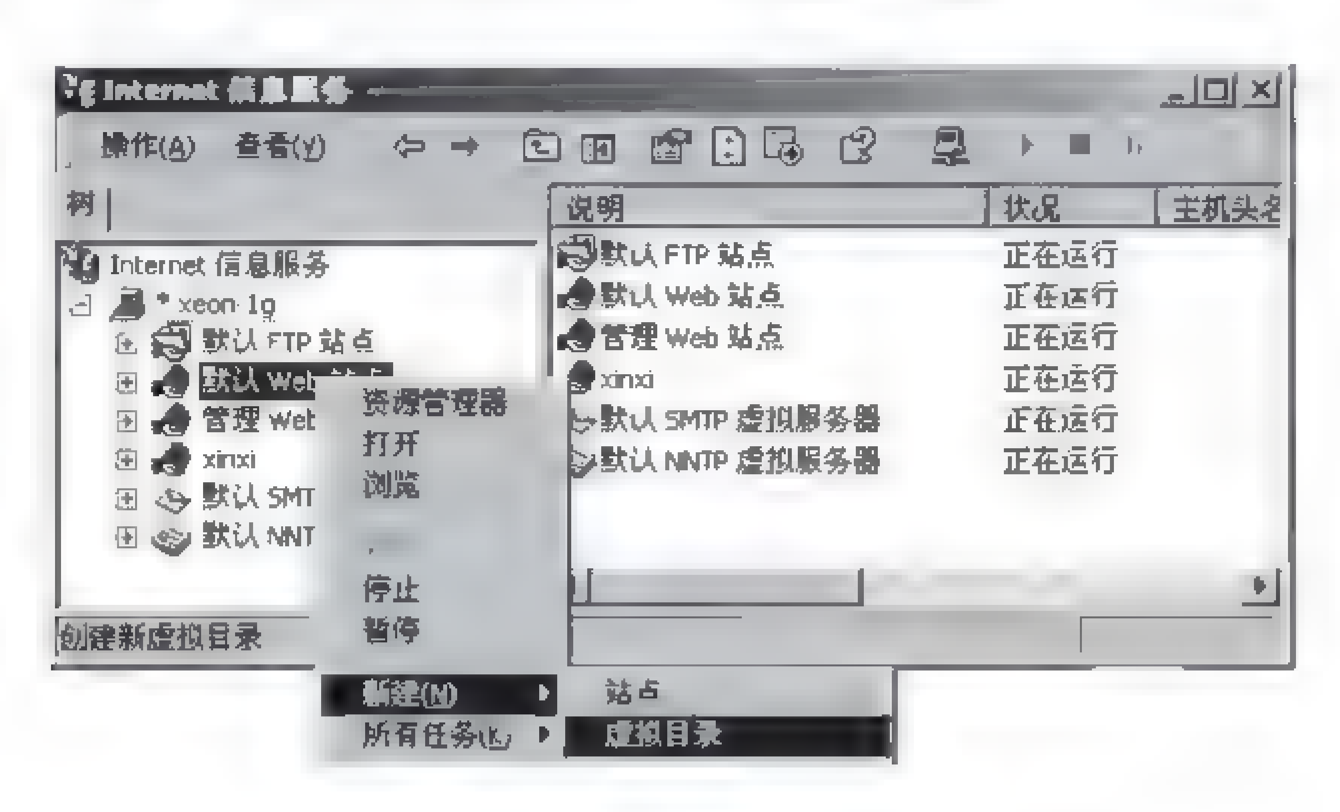


图　10-20

（4）单击“下一步”按钮，在“虚拟目录别名”对话框的“别名”文本框中输入你的别名，如图 10-22 所示。

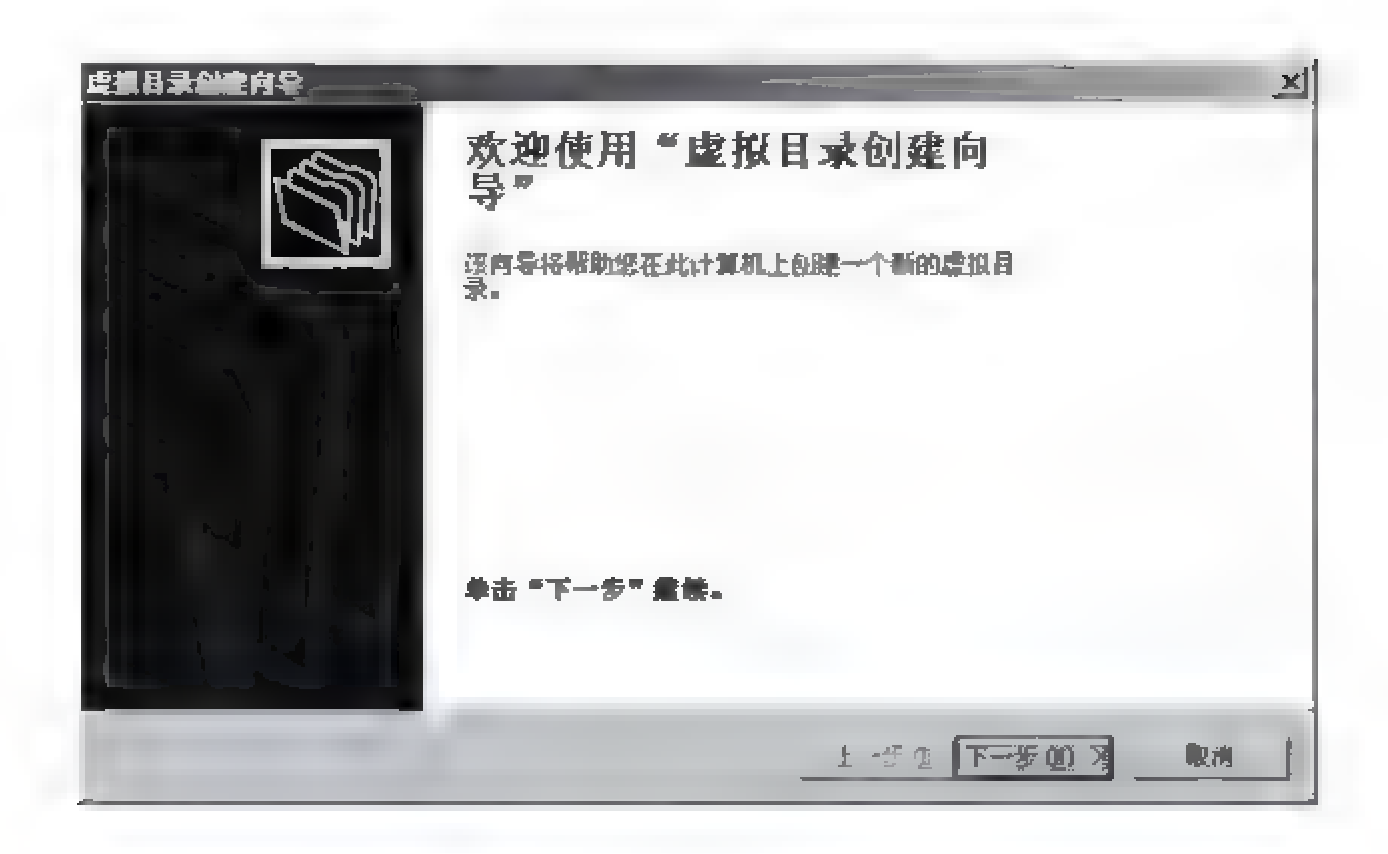


图 10-21

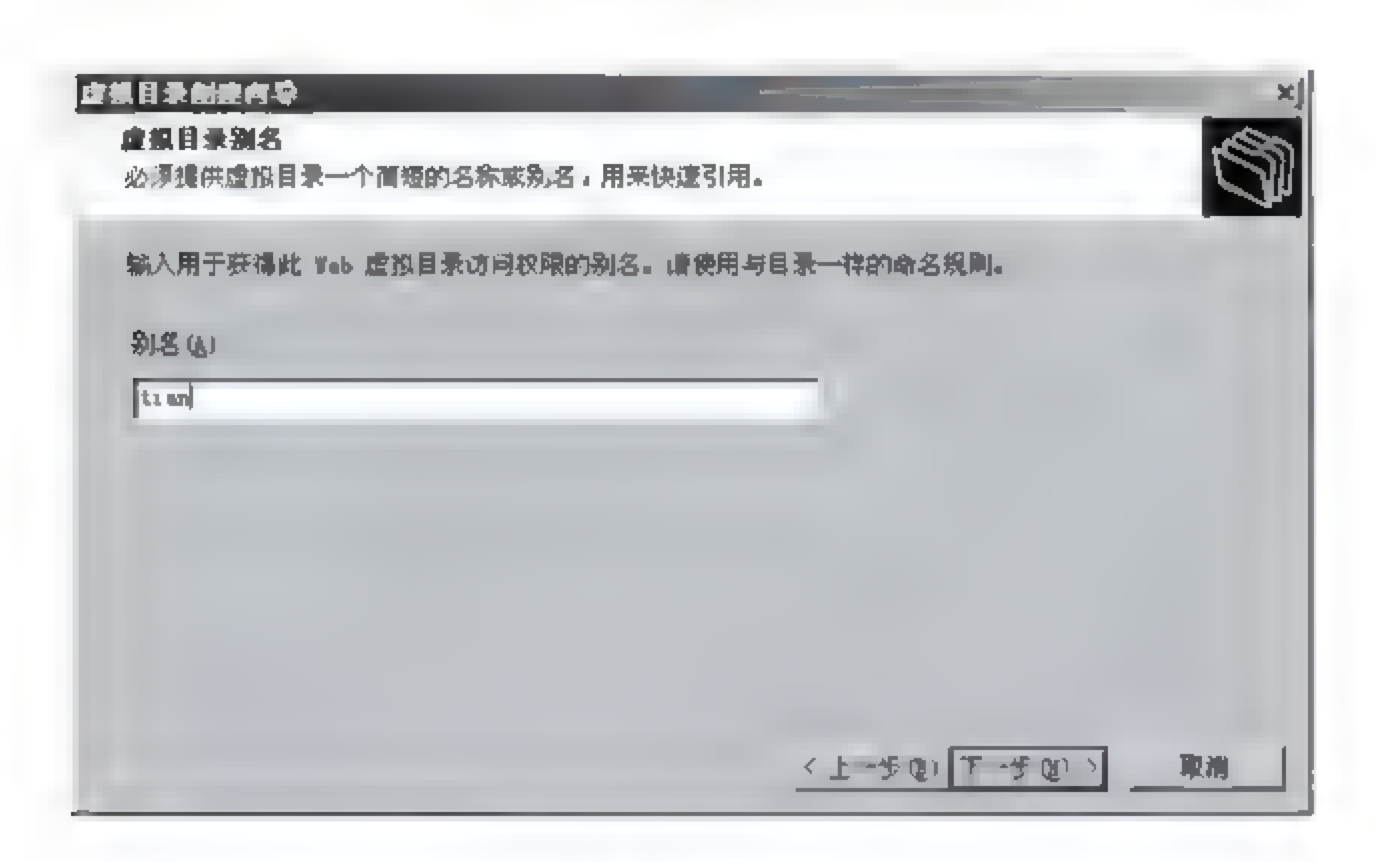


图 10-22

（5）单击“下一步”按钮，出现图 10-23 所示的“Web 站点内容目录”对话框，在“目录”文本框中输入你想发布的 Web 站点的路径或通过“浏览”按钮来查找路径。

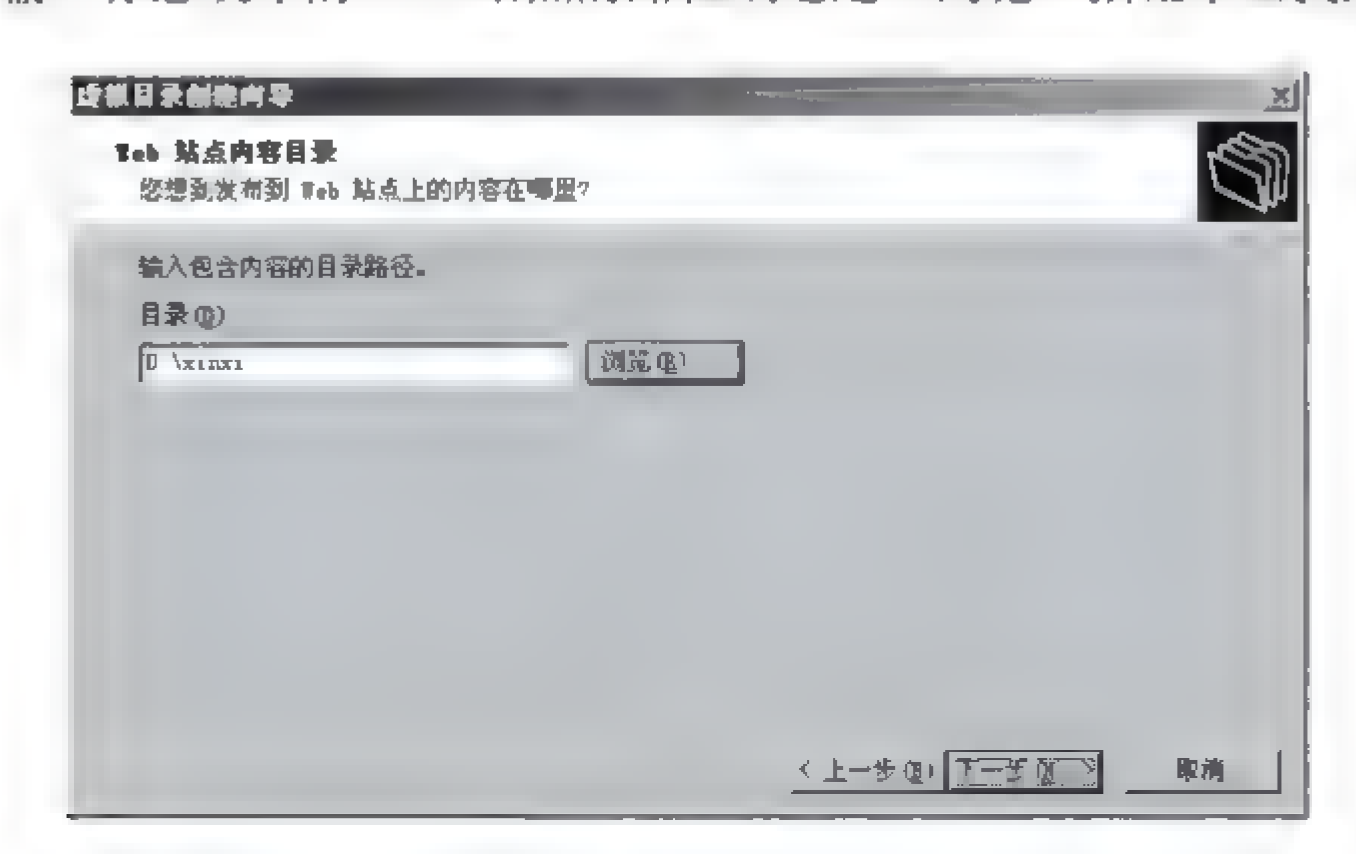


图 10-23

（6）单击“下一步”按钮，出现如图 10-24 所示的“访问权限”对话框。在此对话框中，可对此虚拟目录设置访问权限。

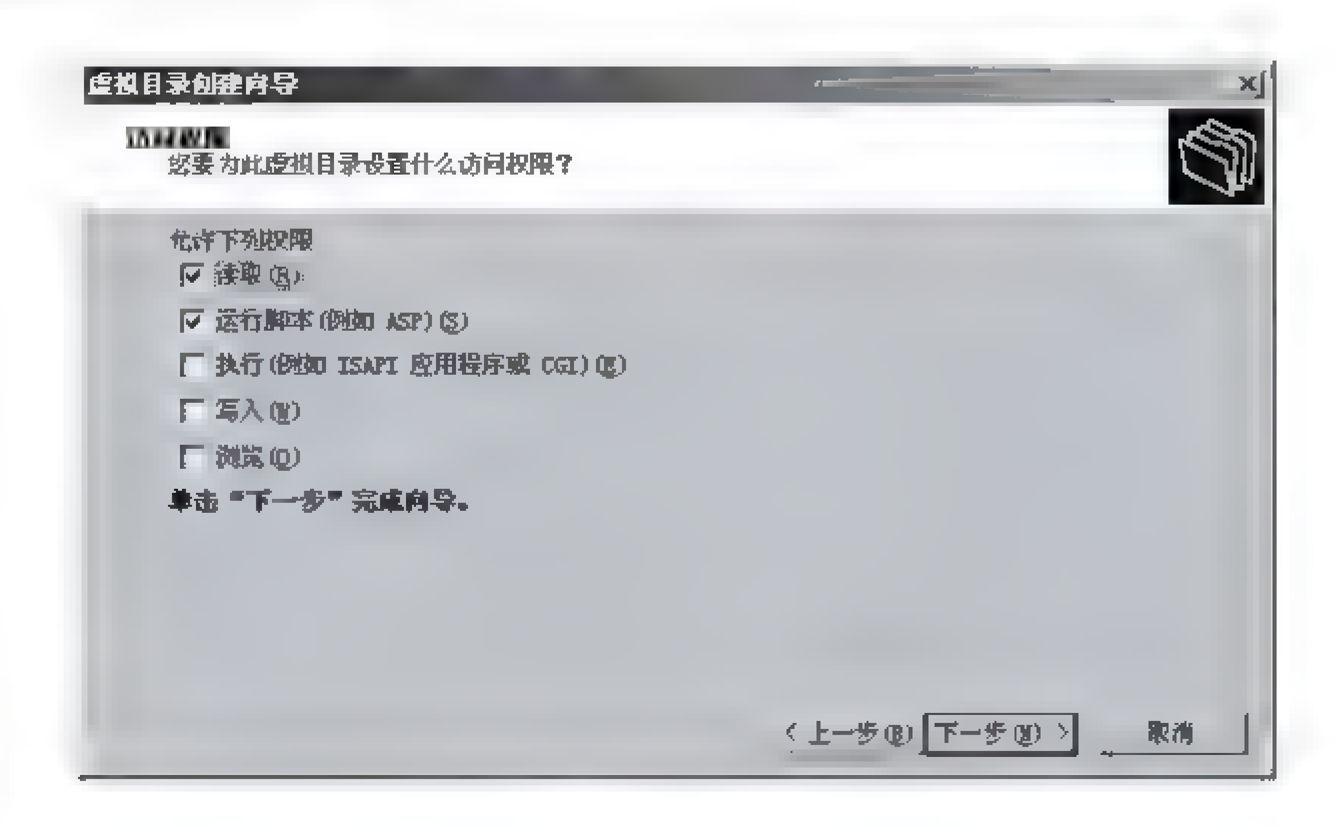


图　10-24

（7）单击“下一步”按钮，出现虚拟目录创建向导对话框，单击“完成”按钮，如图 10-25 所示。



图　10-25

2. 建立密钥对和证书请求

（1）选择“开始”→“程序”→“管理工具”→“Internet 服务管理器”命令，出现图 10-26 所示的“Internet 信息服务”对话框。



图　10-26

（2）双击“计算机”下的 xeon-1g 选项，在“默认 Web 站点”图标处右击，选择“属性”选项，如图 10-27 所示。

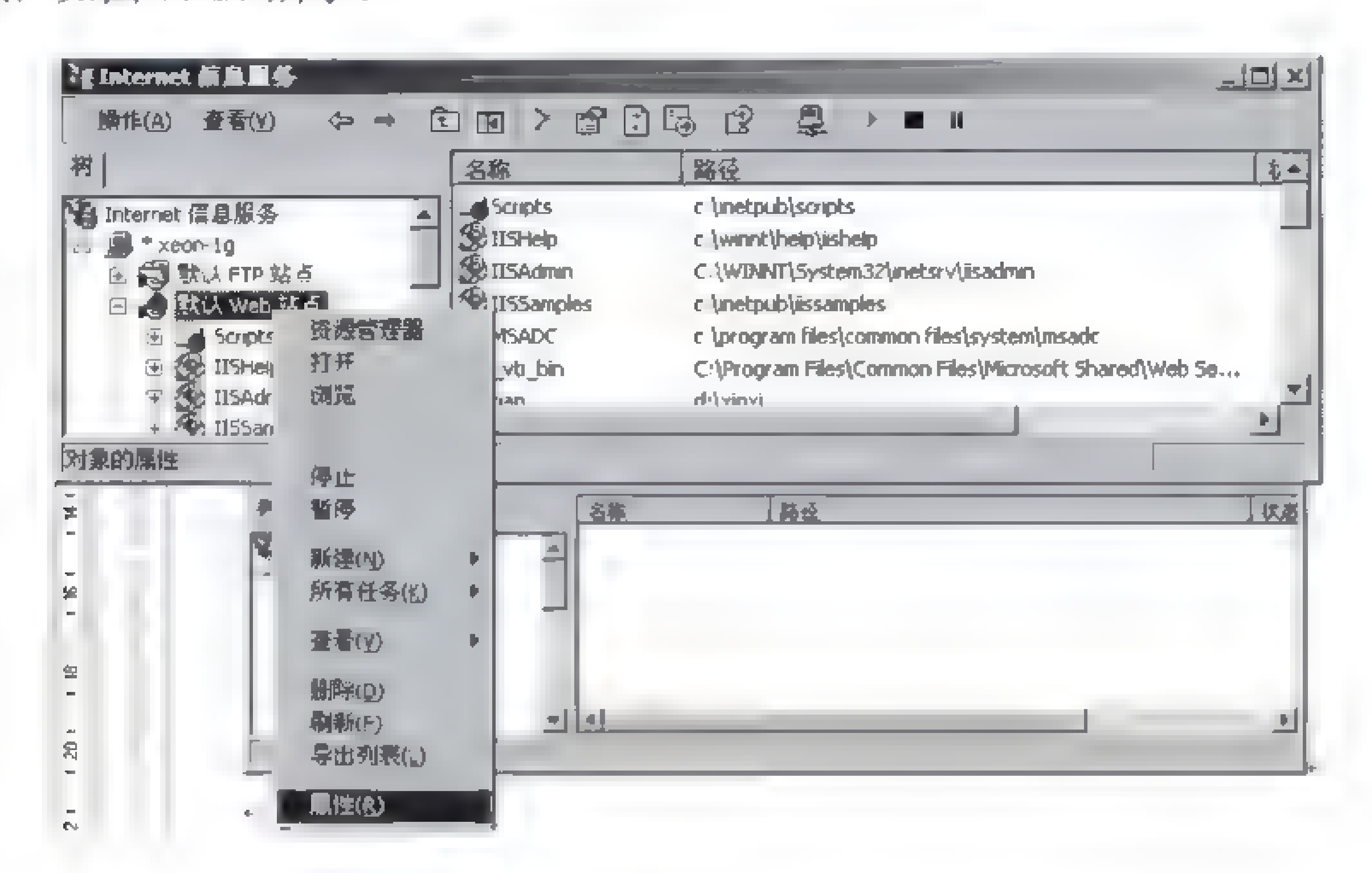


图 10-27

（3）出现“默认 Web 站点属性”对话框，如图 10-28 所示。单击“目录安全性”选项卡，出现图 10-29 所示的对话框。

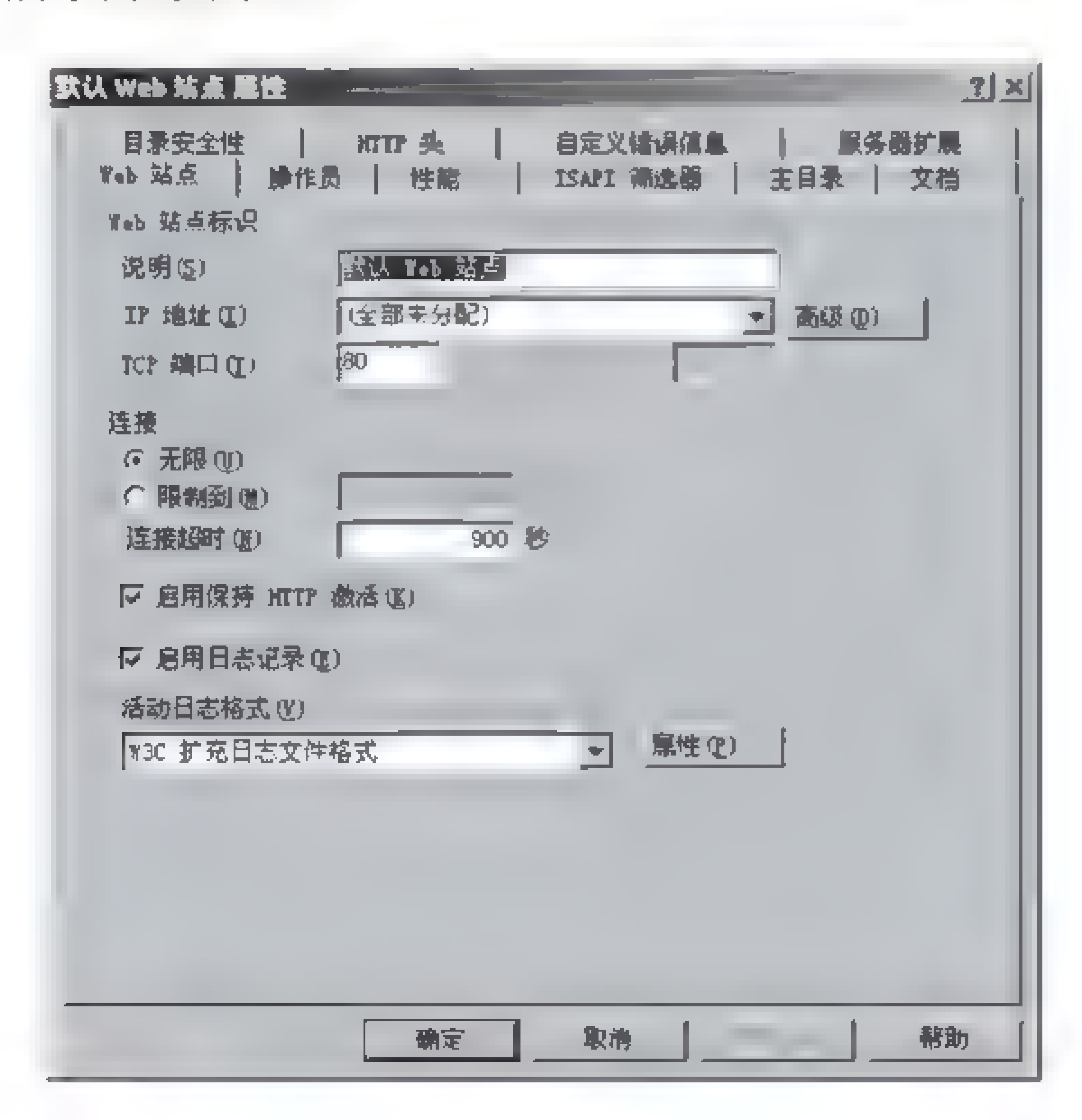


图 10-28

（4）单击“服务器证书”按钮，出现“欢迎使用 Web 服务器证书向导”对话框，如图 10-30 所示。

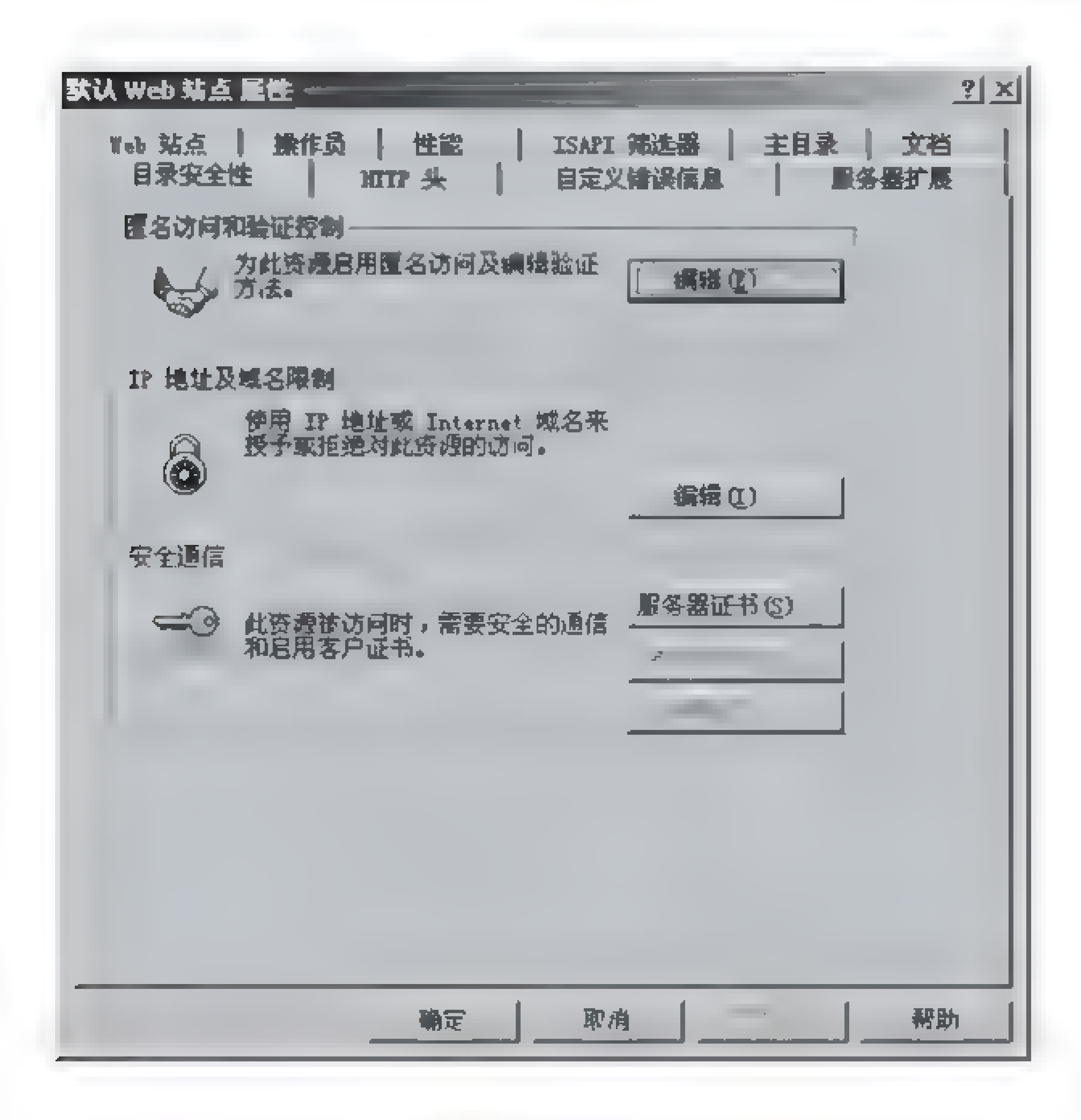


图　10-29

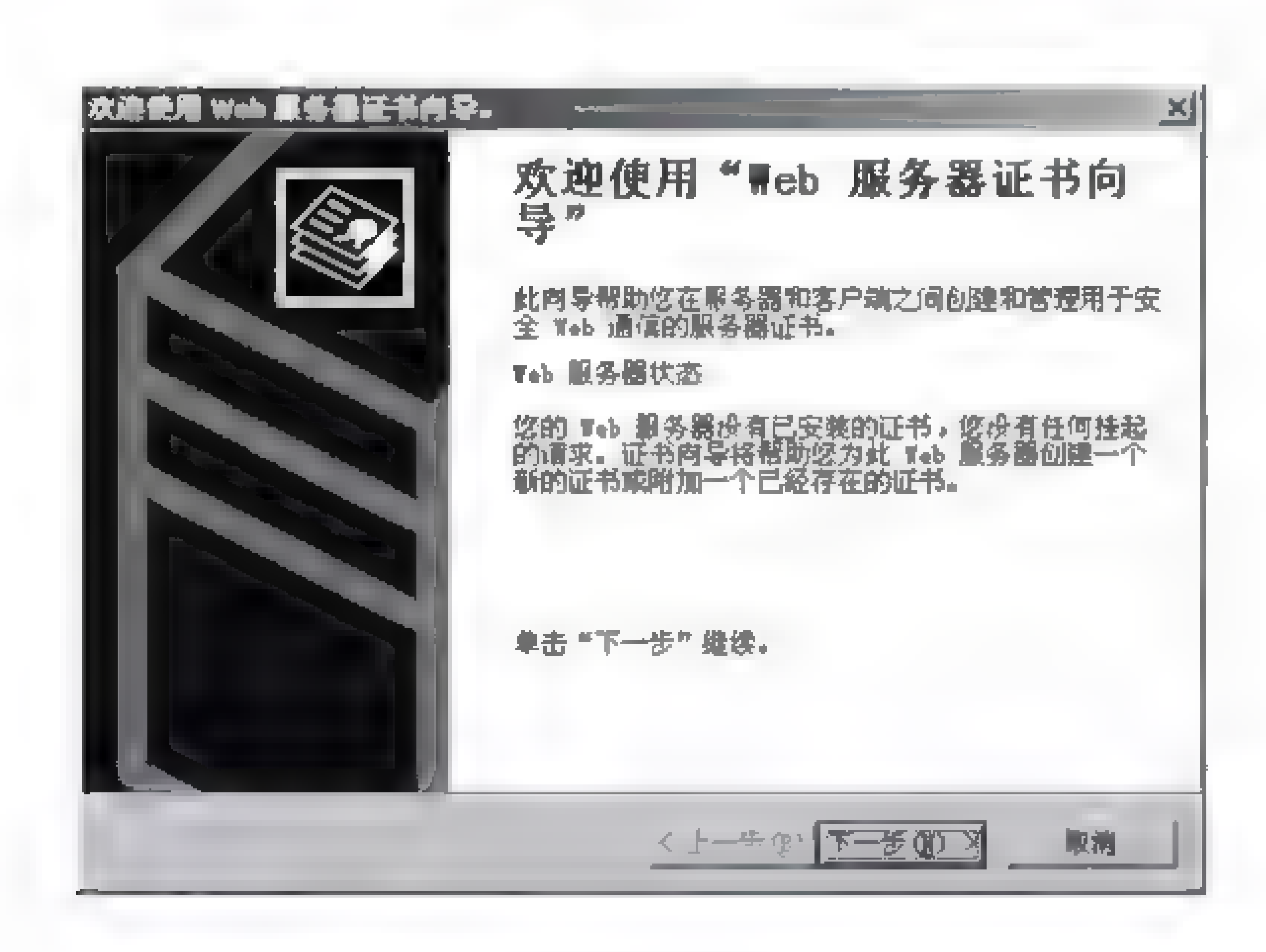


图　10-30

（5）单击“下一步”按钮，出现图 10-31 所示的“服务器证书”对话框，选择“创建一个新证书”单选按钮。

（6）单击“下一步”按钮，选择“现在准备请求，但稍候发送”单选按钮，如图 10-32 所示。

（7）单击“下一步”按钮，在图 10-33 所示的“命名和安全设置”对话框中对新证书进行命名和安全设置，这里均采用默认值。

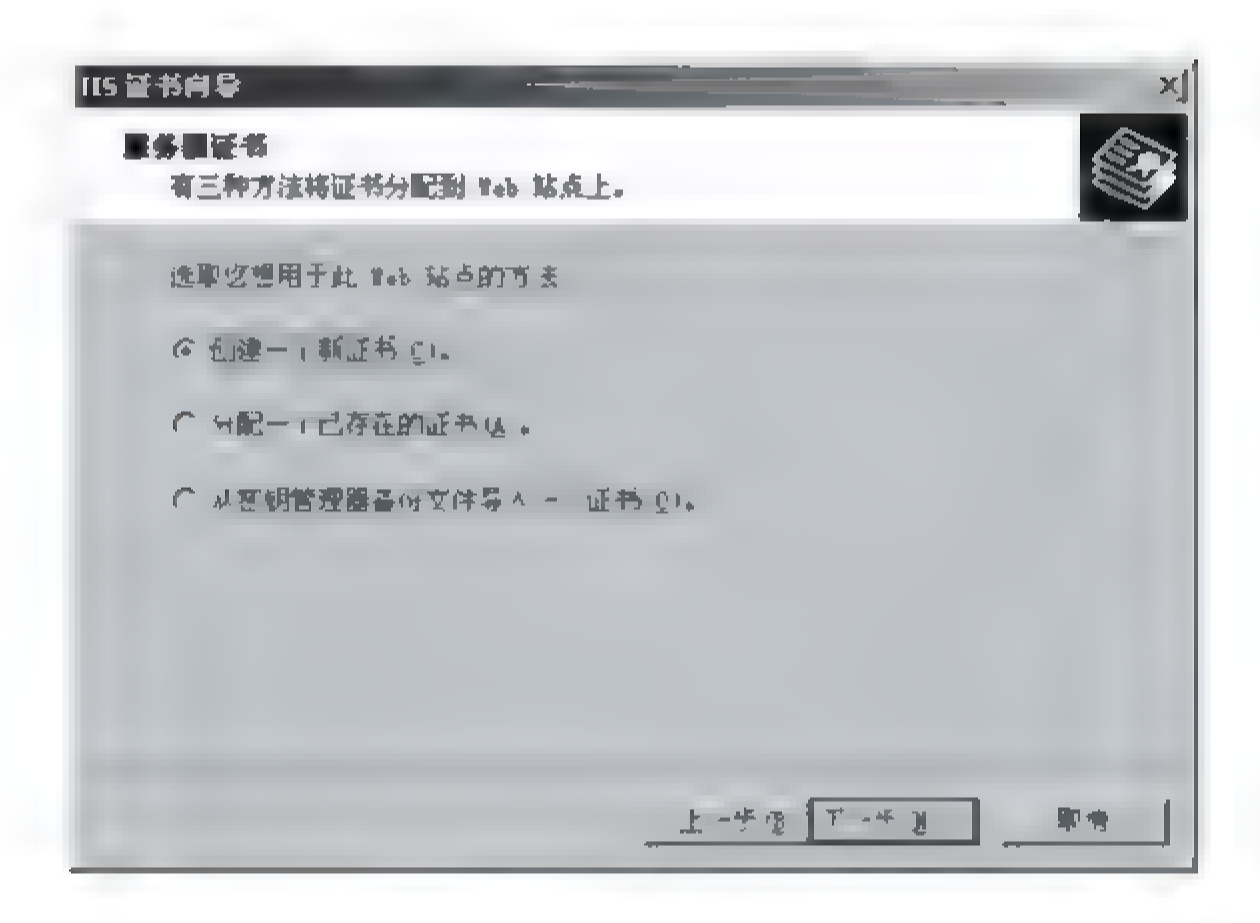


图 10-31

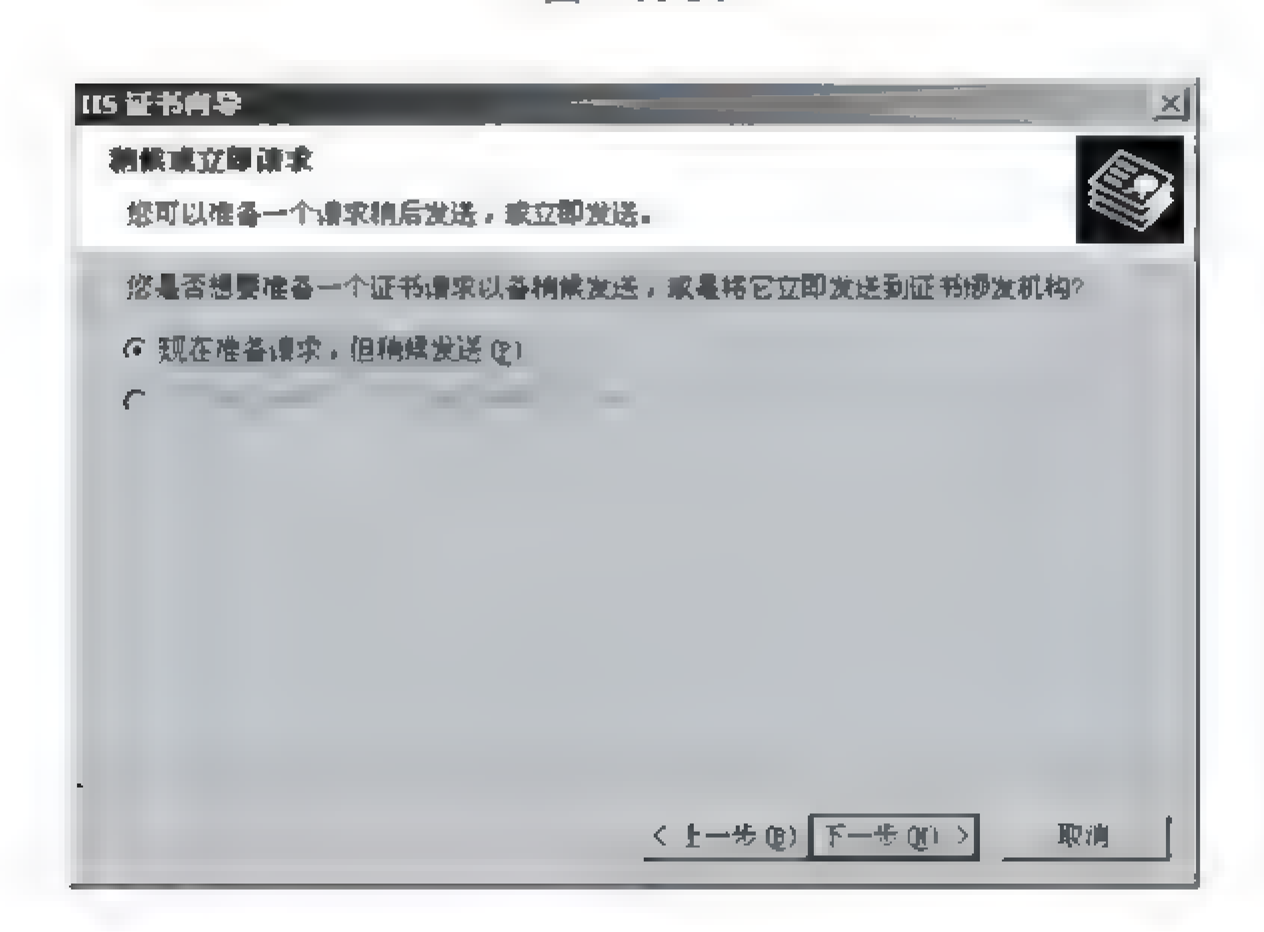


图 10-32

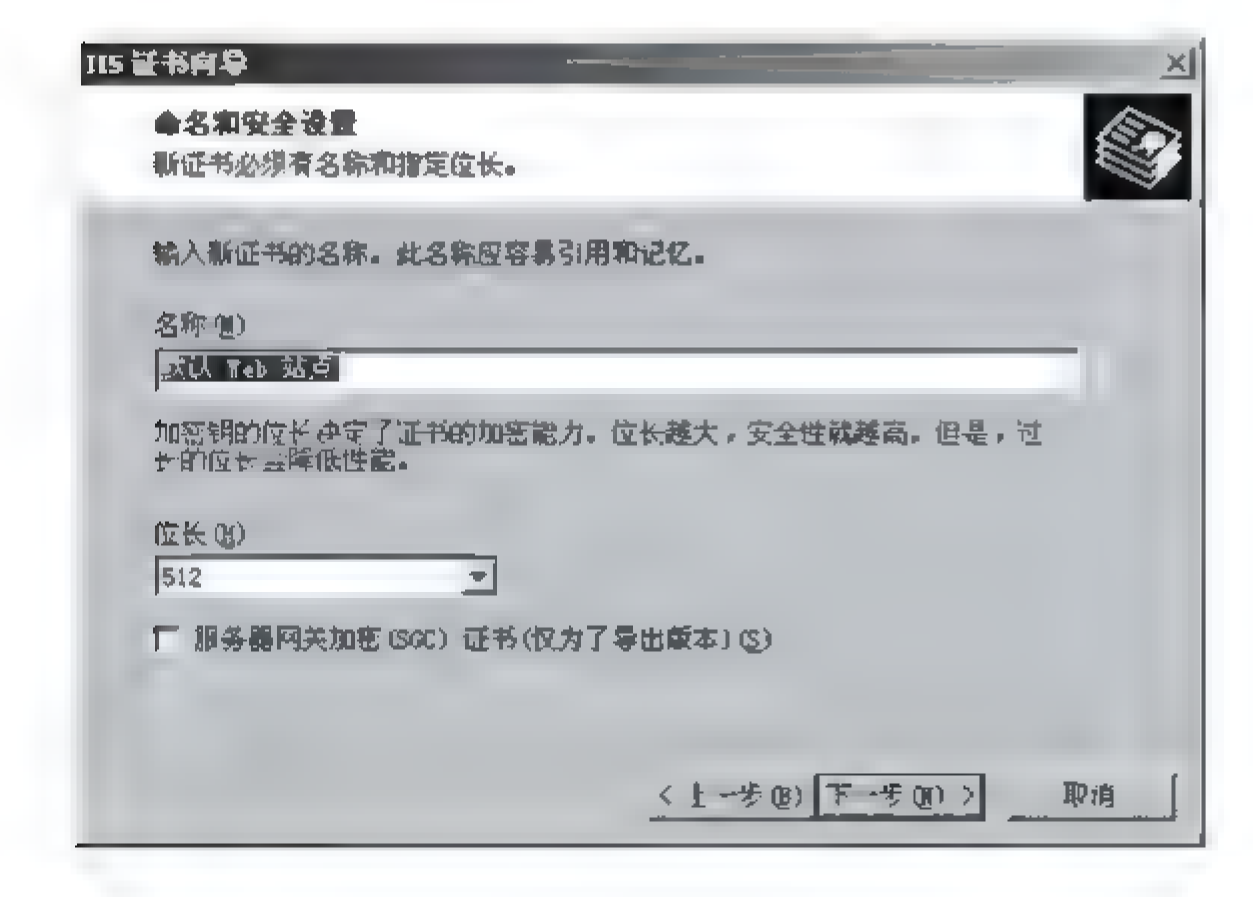


图 10-33

（8）单击“下一步”按钮，在图 10-34 所示的“组织信息”对话框的“组织”和“组织部门”文本框中输入合法的名称。

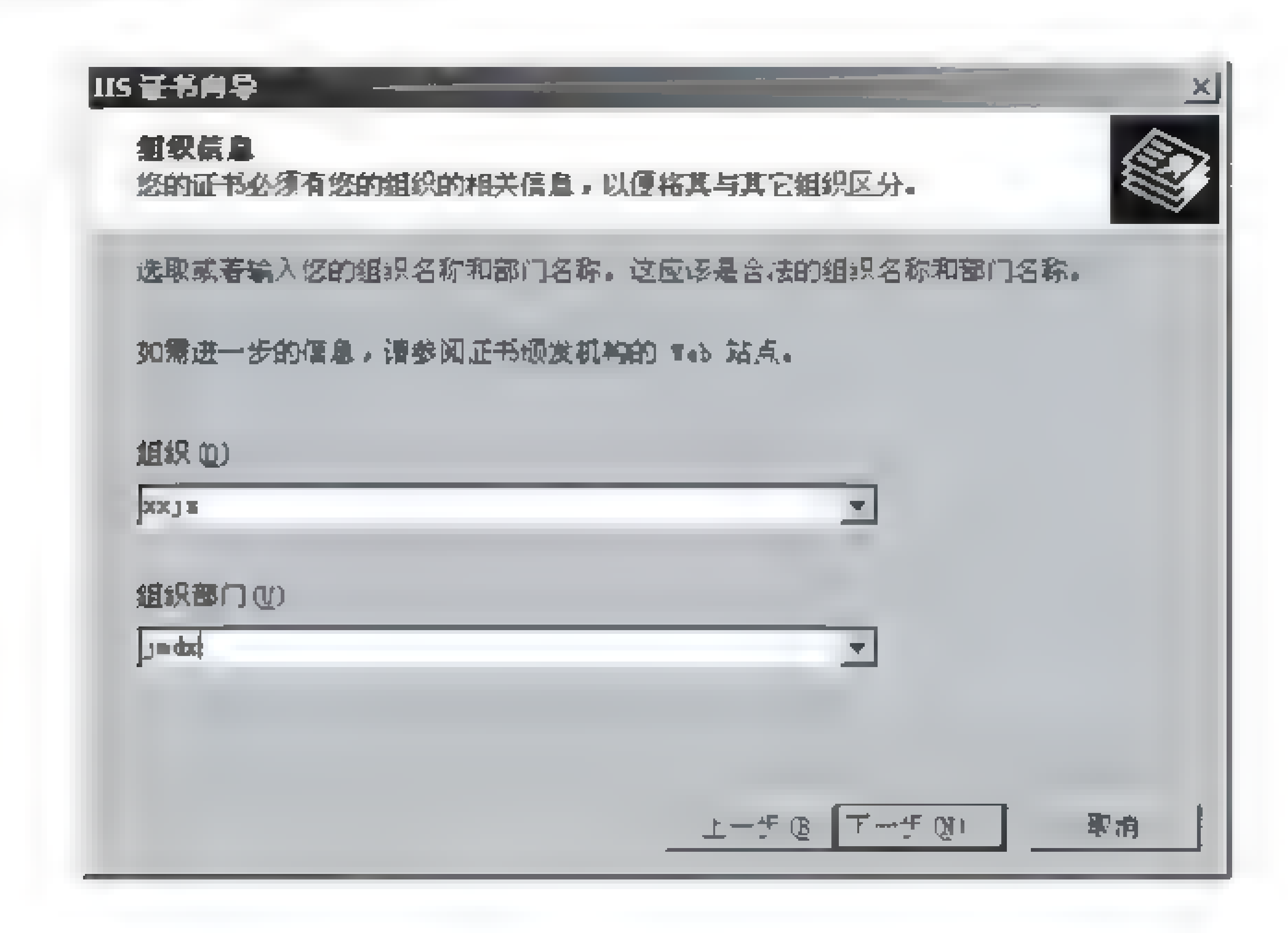


图　10-34

（9）单击“下一步”按钮，在图 10-35 所示的“站点的公用名称”对话框中输入站点的公用名称。



图　10-35

（10）单击“下一步”按钮，在图 10-36 所示的“地理信息”对话框中输入证书的地理信息。

（11）单击“下一步”按钮，在图 10-37 所示的“证书请求文件名”对话框中为证书请求输入一个文件名。

（12）单击“下一步”按钮，生成“请求文件摘要”对话框，如图 10-38 所示。

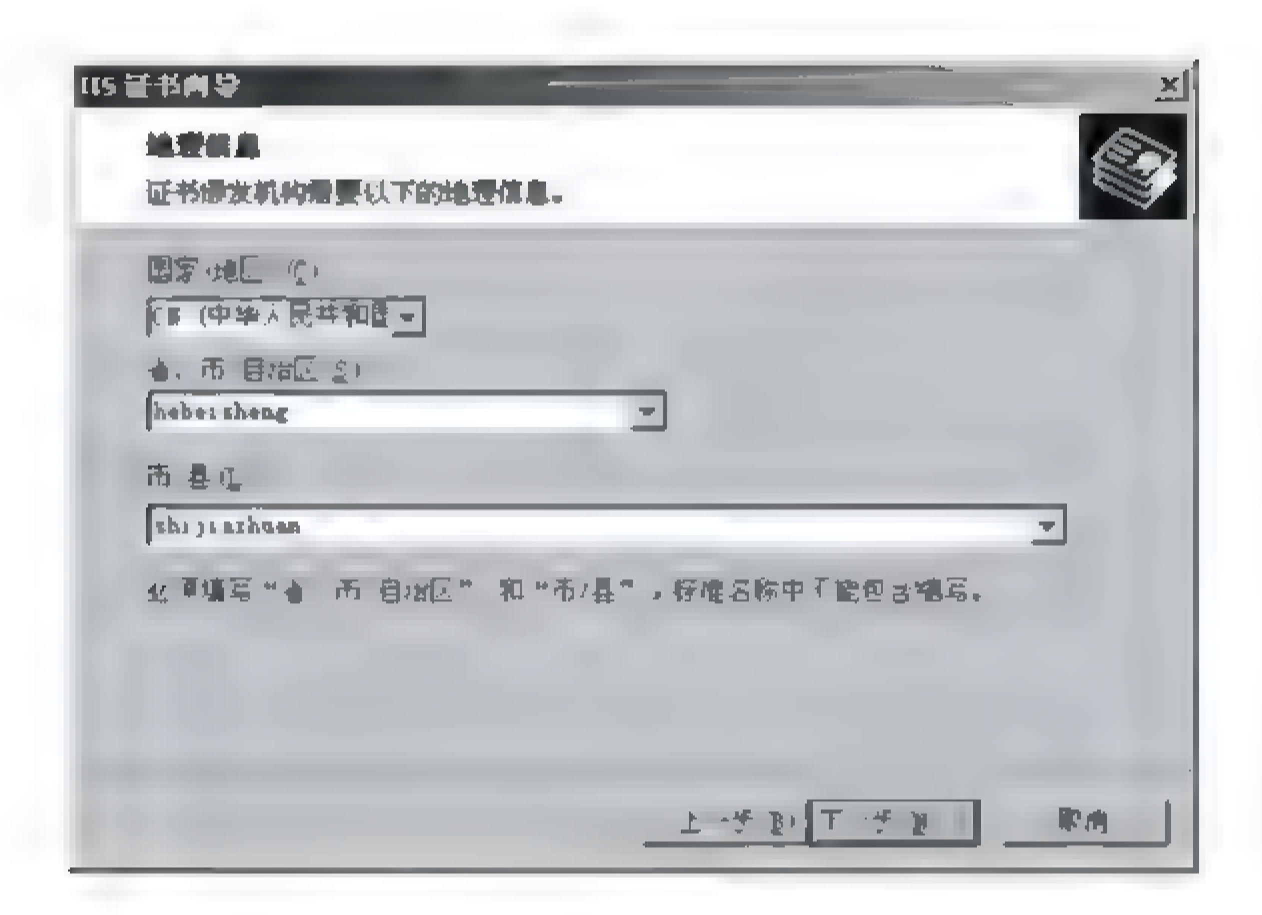


图 10-36

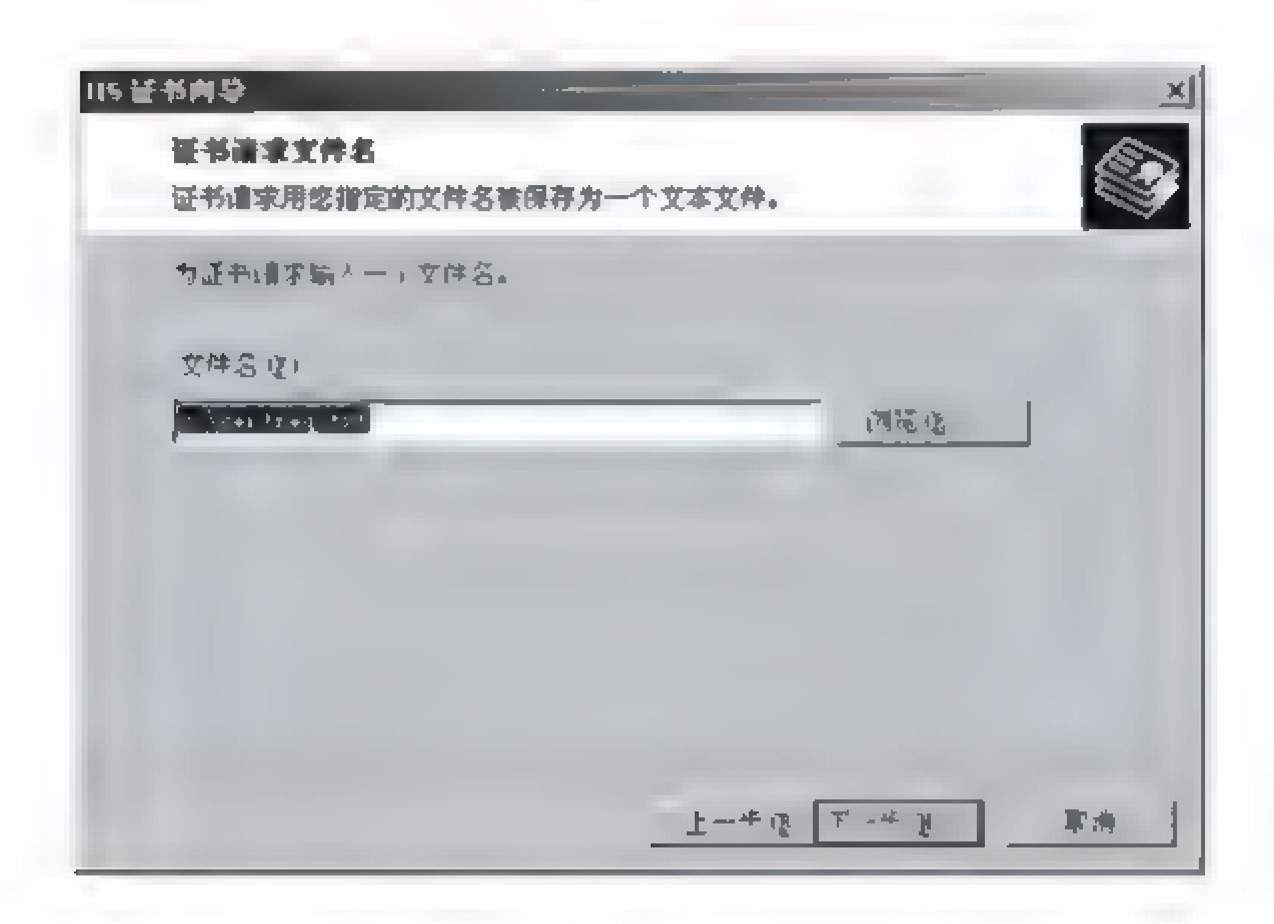


图 10-37

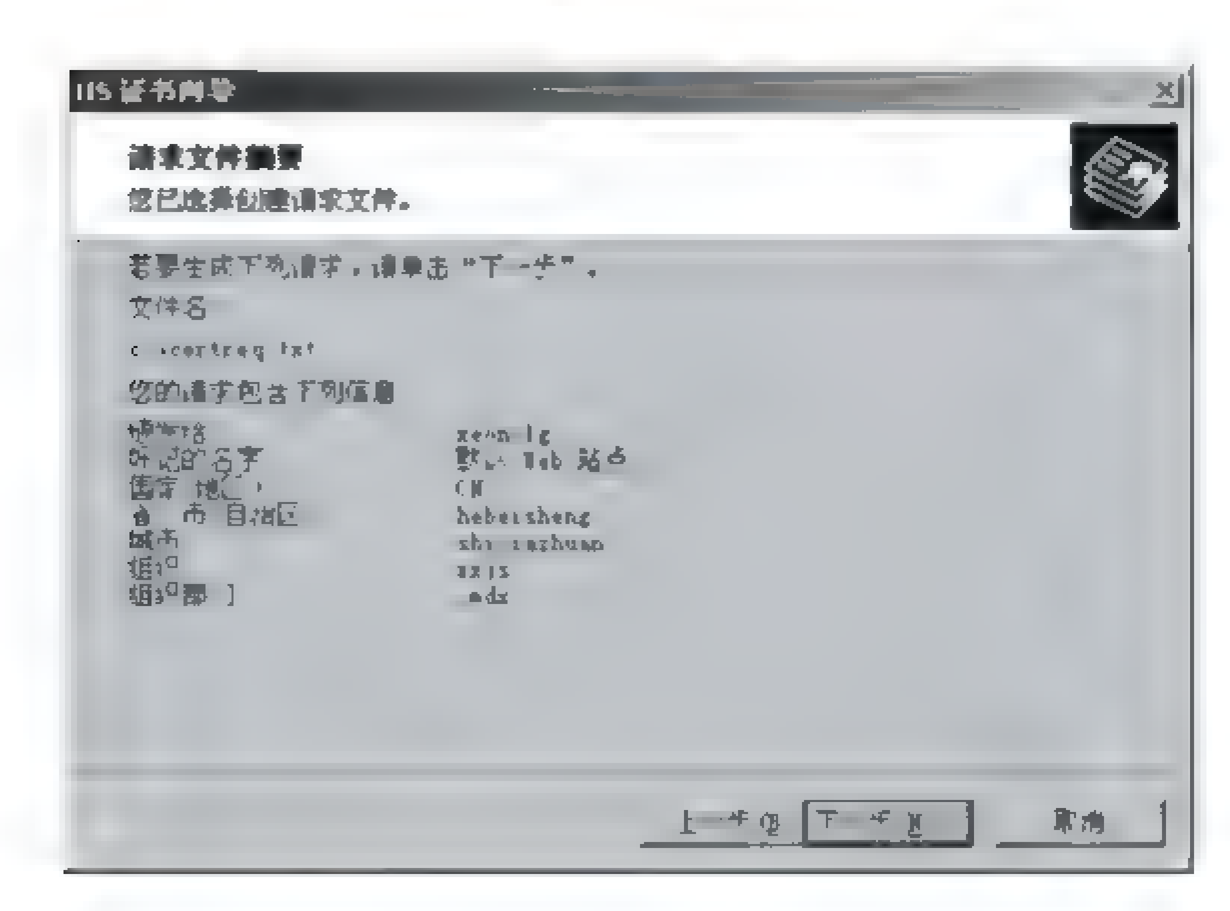


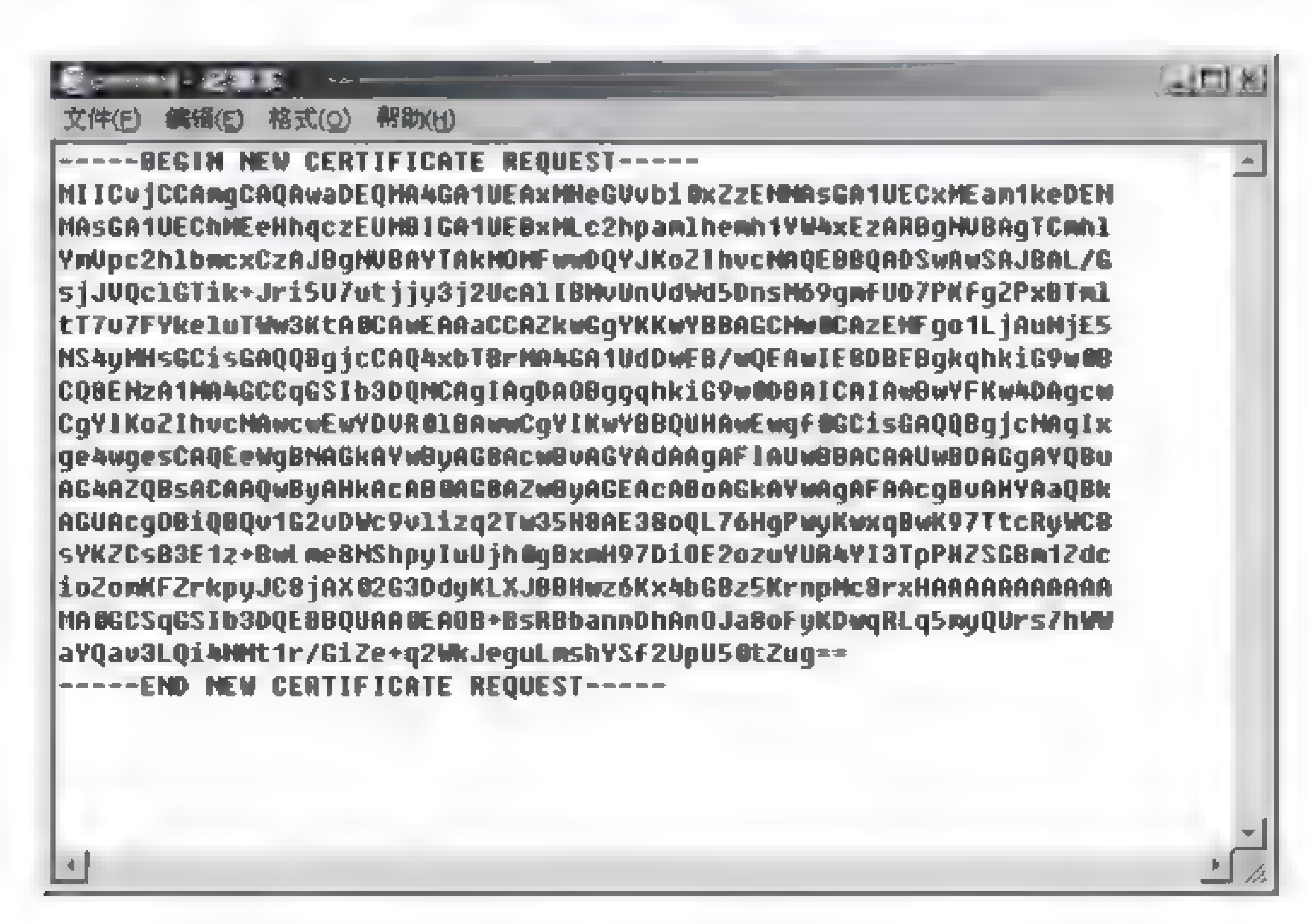
图 10-38

（13）单击“下一步”按钮，出现“完成 Web 服务器证书向导”对话框，单击“完成”按钮，如图 10-39 所示。



图　10-39

至此，完成了 Web 服务器证书请求。新建的证书请求文件存放在 C:\certreq.txt 中，包含新的加密过的证书请求字符串，如图 10-40 所示。



-----BEGIN NEW CERTIFICATE REQUEST-----
MIICvjCCAmgCAQAwaDEQMA4GA1UEAxMHeGVvb10xZzEMMAsGA1UECxMEam1keDEN
MAsGA1UEChMEeHhqczEUMBIGA1UEBxMLc2hpamlhemh1YW4xEzARBgNVBAgTCmhl
YmVpc2hlbmcxCzAJBgNVBAYTAkNOMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAL/G
sjJVQclGTik+JriSU7utjjy3j2UcAlIBMvUnVdWd5DnsM69gmFU07PKFg2PxBTml
tT7u7FYkeluTWw3KtA0CAwEAAaCCAZkwGgYKKwYBBAGCNw0CAzEMFgo1LjAuMjE5
NS4yMHsGCisGAQQBgjcCAQ4xbTBrMA4GA1UdDwEB/wQEAwIE8DBEBgkqhkiG9w0B
CQ8ENzA1MA4GCCqGSIb3DQMCAgIAgDAOBggqhkiG9w0DBAICAIAwBwYFKw4DAgcw
CgYIKoZIhvcNAwcwEwYDVR0lBAwwCgYIKwYBBQUHAwEwgf0GCisGAQQBgjcNAgIx
ge4wgesCAQEeWgBNAGkAYwByAG8AcwBvAGYAdAAgAFIAUwBBACAAUwBDAGgAYQBu
AG4AZQBsACAAQwByAHkAcAB0AG8AZwByAGEAcABoAGkAYwAgAFAAcgBvAHYAaQBk
AGUAcgOBiQBQv1G2vDWc9vlizq2Tw35H8AE38oQL76HgPwyKwxqBwK97TtcRyWC8
sYKZCsB3E1z+Bwlme8NShpyIuUjh0gBxmH97DiOE2ozuYUR4YI3TpPHZSGBm1Zdc
ioZomKFZrkpyJC8jAX02G3DdyKLXJ0BHwz6Kx4bGBz5KrnpMc8rxHAAAAAAAAAAA
MA0GCSqGSIb3DQEBBQUAA0EA0B+BsRBbannDhAnOJa8oFyKDwqRLq5myQUrs7hWW
aYQav3LQi4NMt1r/GiZe+q2WkJeguLmshYSf2UpU50tZug==
-----END NEW CERTIFICATE REQUEST-----

图　10-40

3. 向证书授权机构提交证书请求文件

在完成了密钥对和请求文件之后，需要从一个可信赖的第三方机构请求服务器证书。具体操作步骤如下。

（1）在 IE 浏览器“地址”文本框输入 http://localhost/certsrv，如图 10-41 所示。

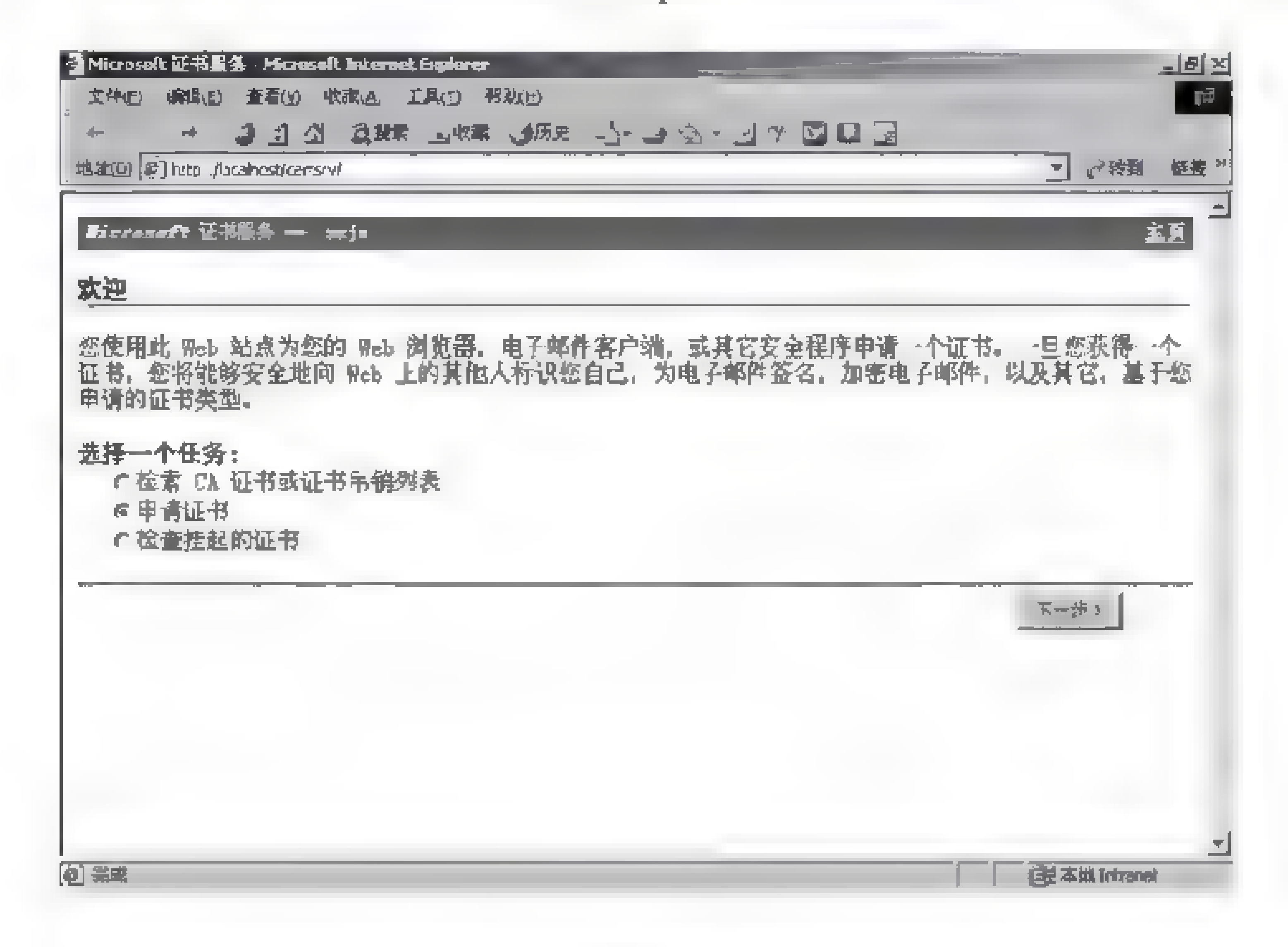


图 10-41

（2）选择申请证书选项，然后单击“下一步”按钮，出现图 10-42 所示的“选择申请类型”对话框，单击“高级申请”单选按钮。

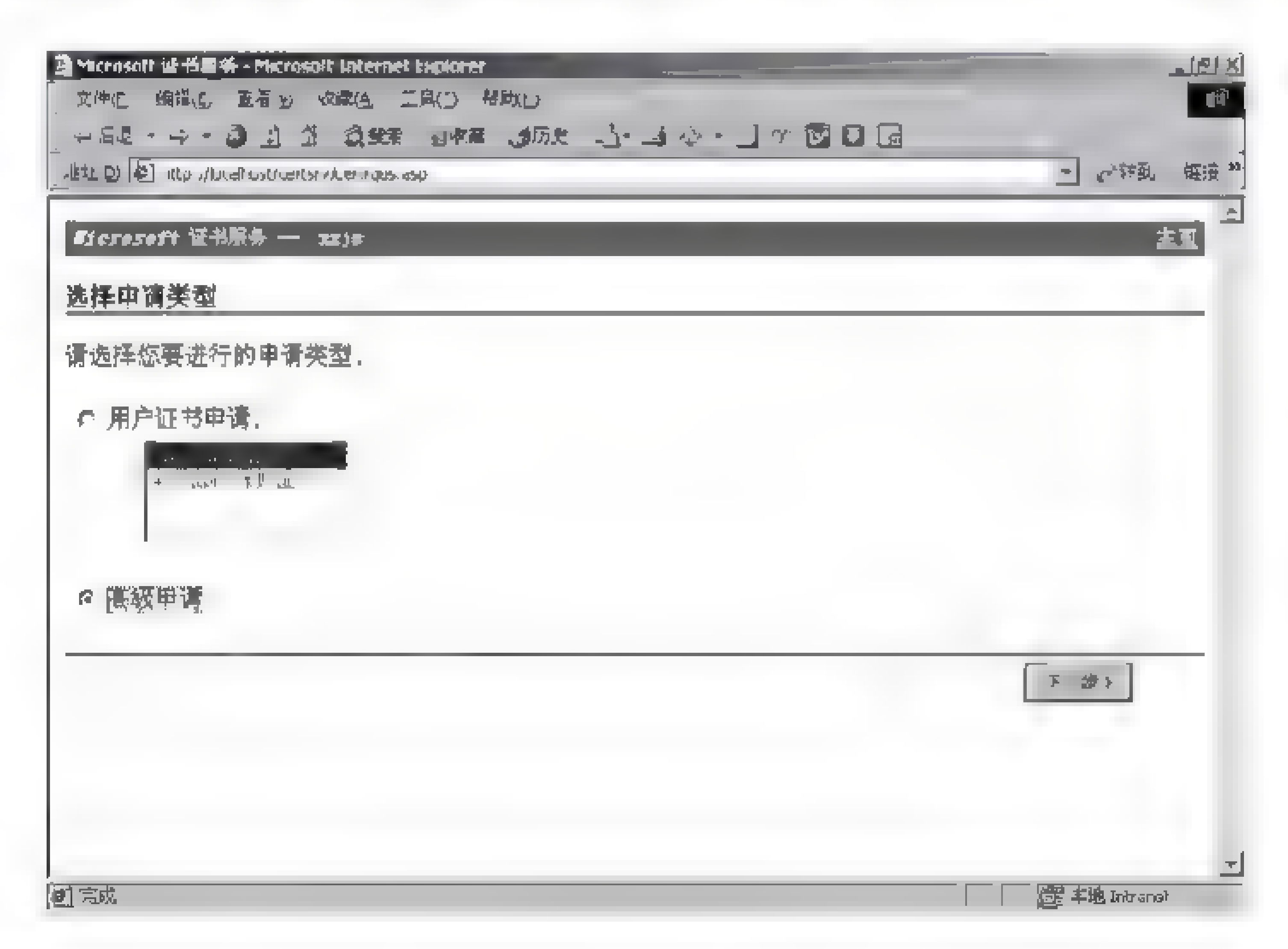


图 10-42

（3）单击“下一步”按钮，在图 10-43 的“高级证书申请”对话框中选择第二项“使用 base64 编码的 PKCS#10 文件提交一个证书申请，或使用 base64 编码的 PKCS #7 文件更新证书申请。

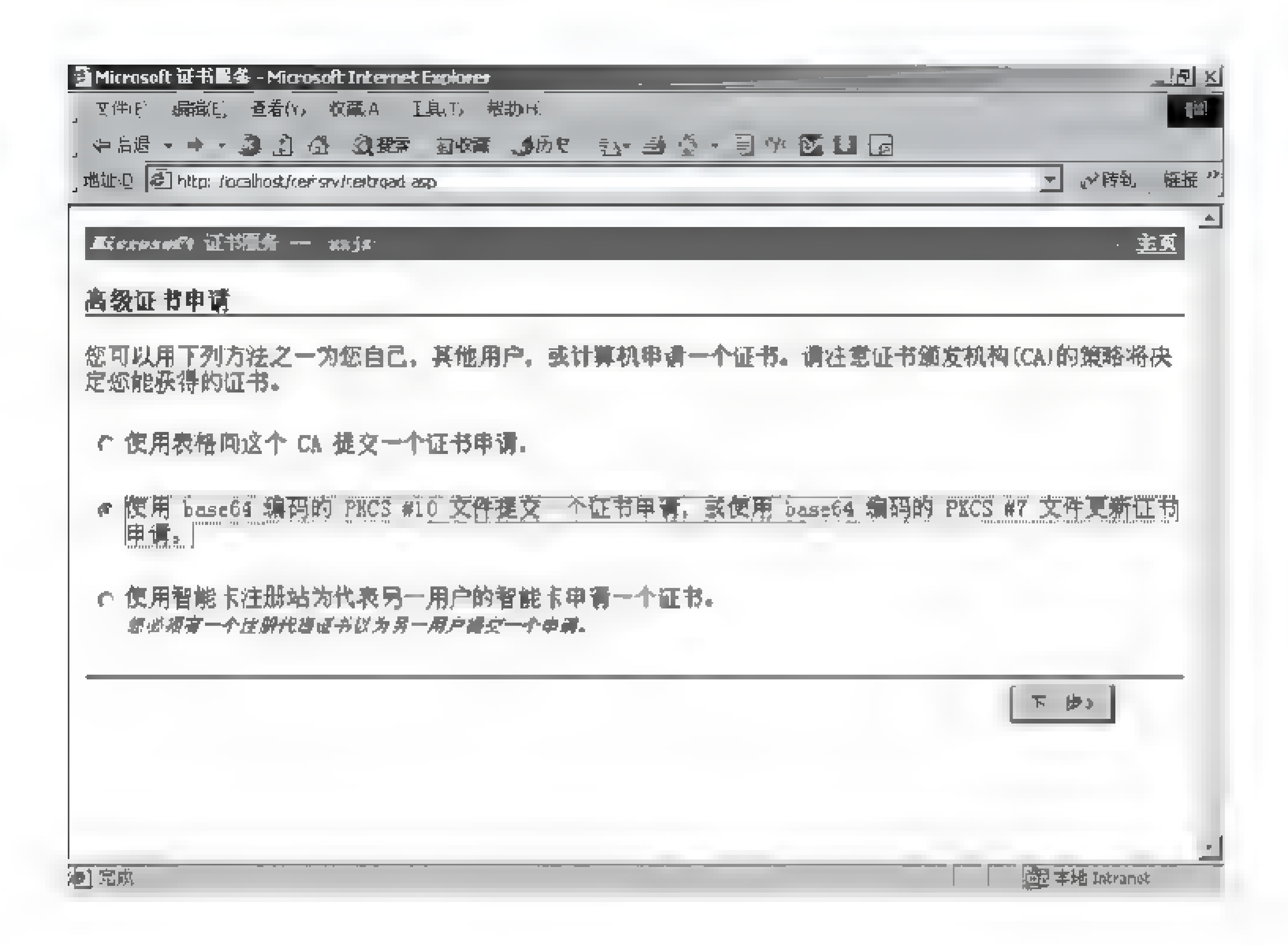


图 10-43

（4）单击“下一步”按钮，将新生成的 C:\certreq.tet 在记事本中打开，将请求文件的内容用 Ctrl+C 组合键复制到剪贴板上，然后粘贴到图 10-44 所示的“提交一个保存的申请”对话框的“保存的申请”文本框中。

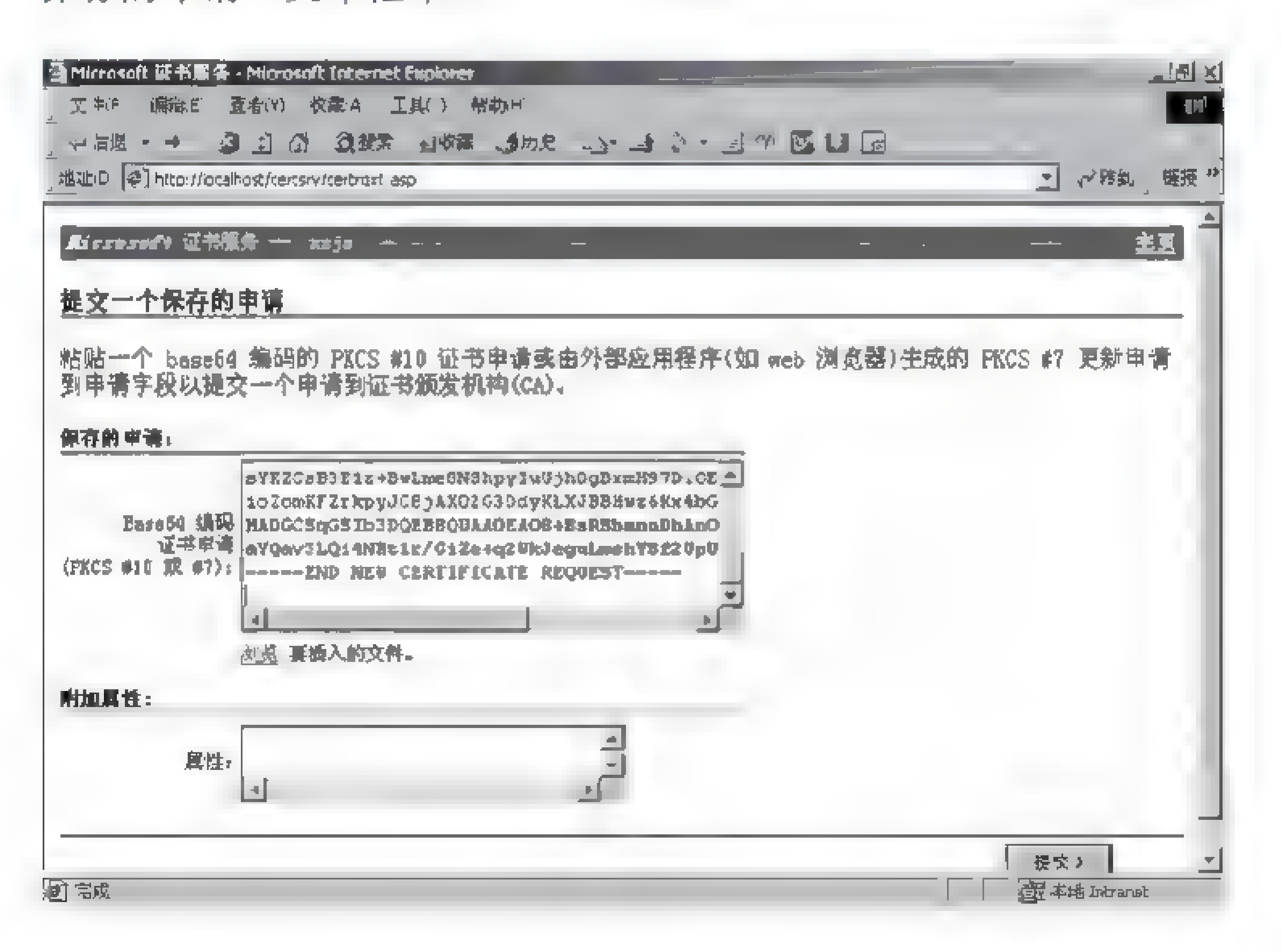


图 10-44

（5）单击“提交”按钮，出现图 10-45 所示的“证书挂起”对话框，表明证书已经收到，等待证书授权机构批准。

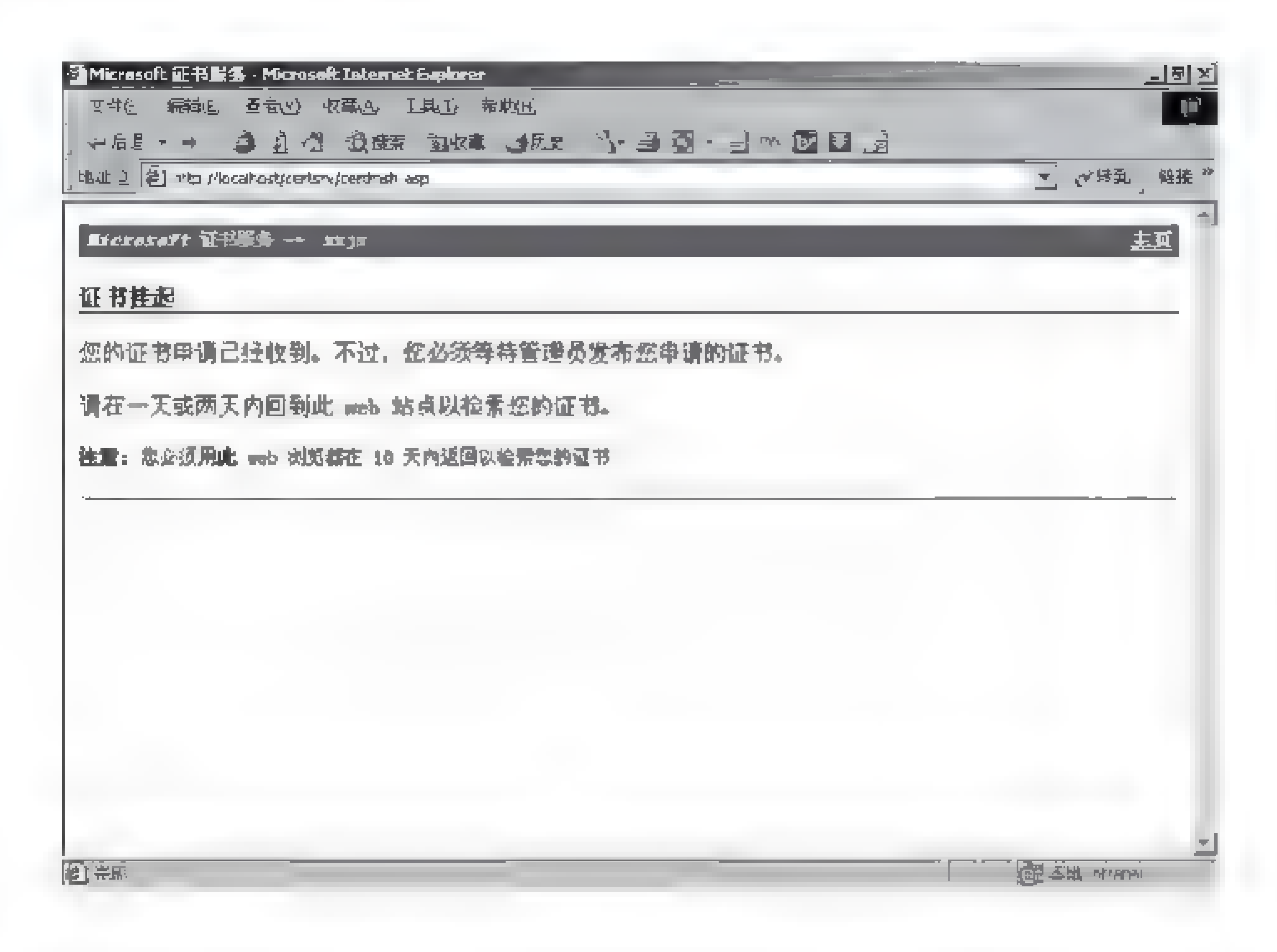


图 10-45

4. 证书服务器工具

（1）选择“开始”→“程序”→“管理工具”→“证书颁发机构”命令，出现图 10-46 所示的证书颁发机构对话框。

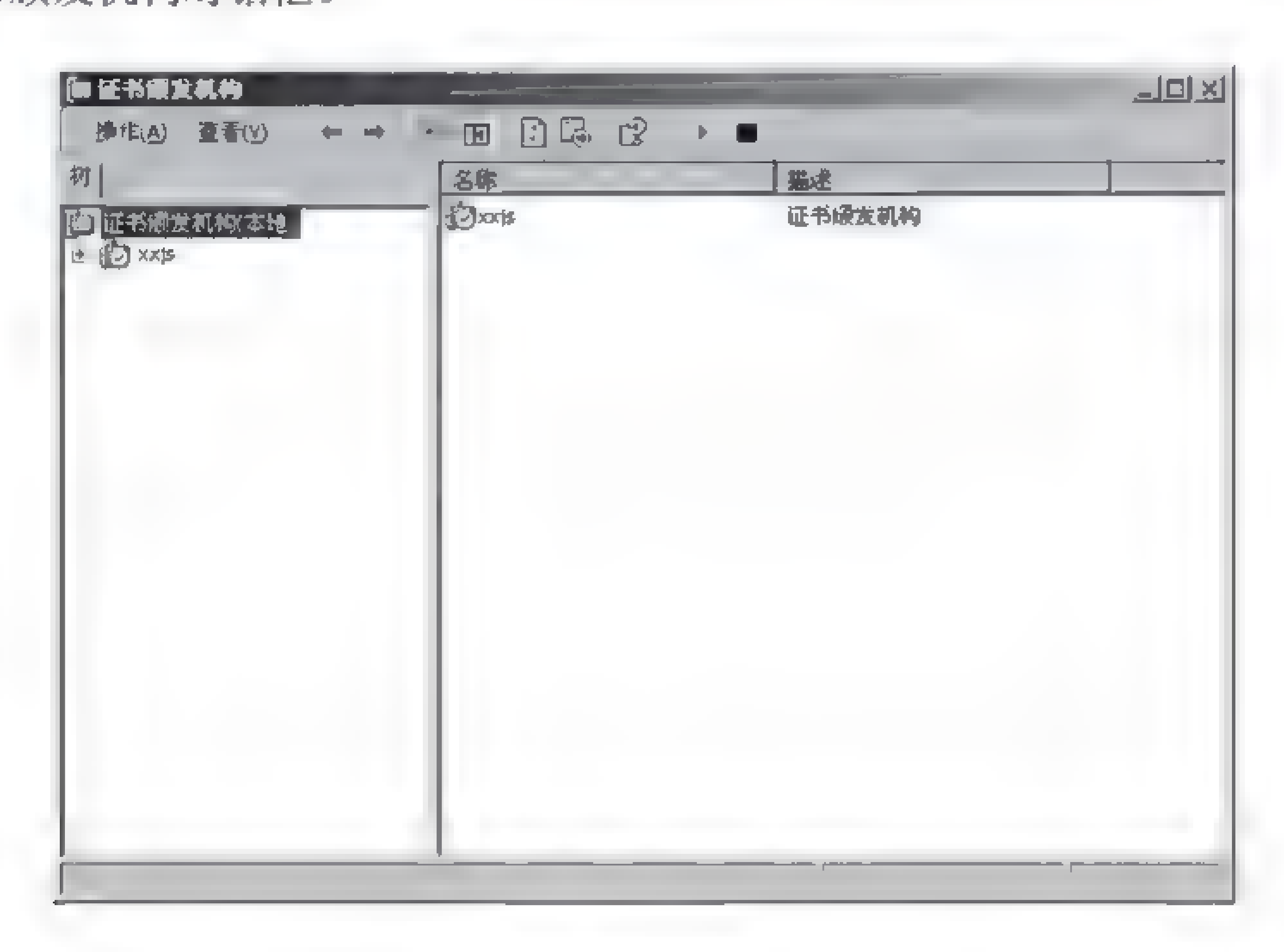


图 10-46

（2）双击 xxjs，在此 root 授权机构下有 4 个选项，如图 10-47 所示。

- 吊销的证书：包含所有被发布又被撤销的证书。
- 颁发的证书：包含所有被批准并颁发的证书。

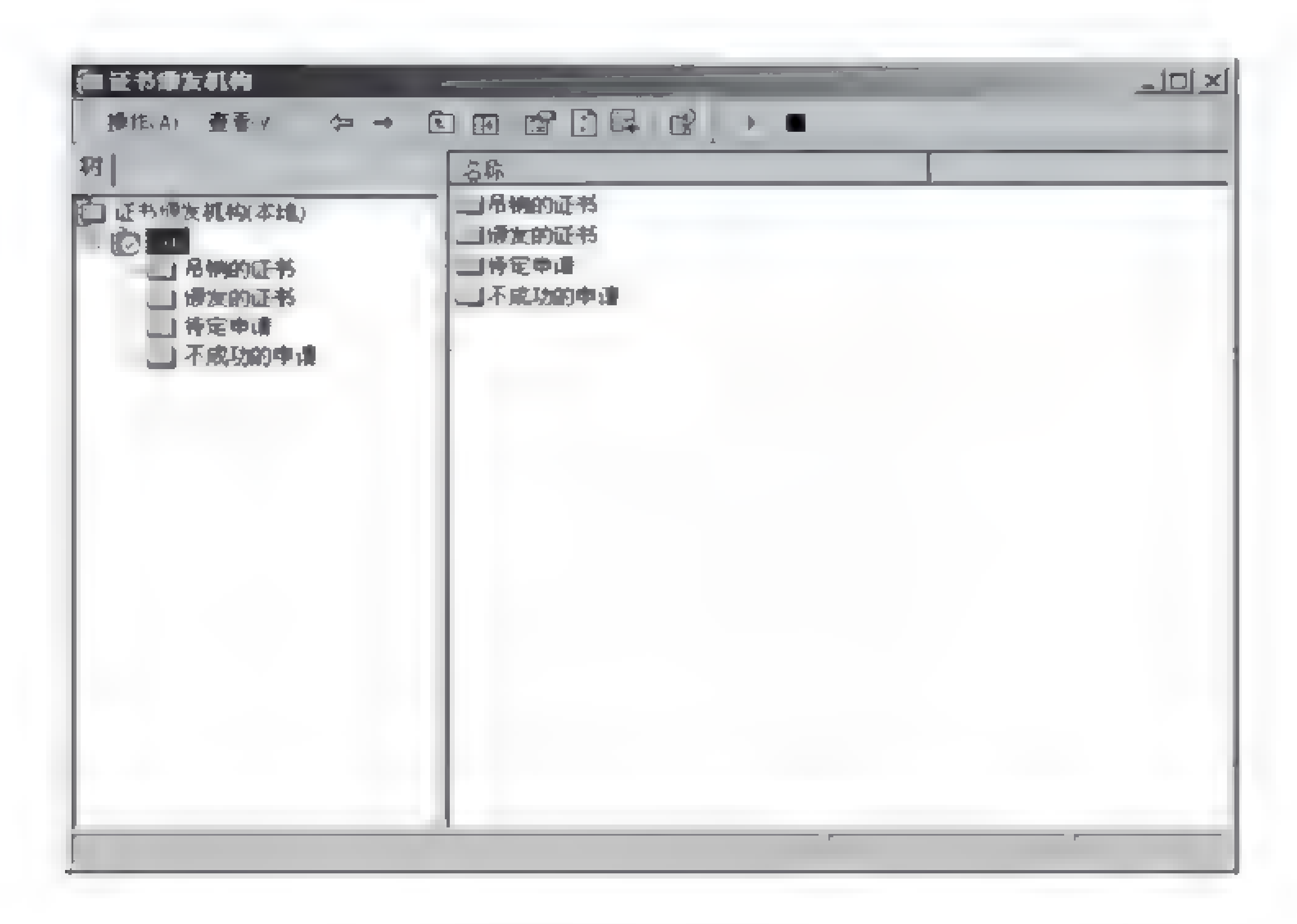


图　10-47

- 待定申请：包含了所有等待 root 授权机构批准的证书请求。
- 不成功的申请：包含了所有被拒绝的证书请求。

（3）选择第三项“待定申请”，如图 10-48 所示，在右窗口可以看到所提交的证书请求文件。

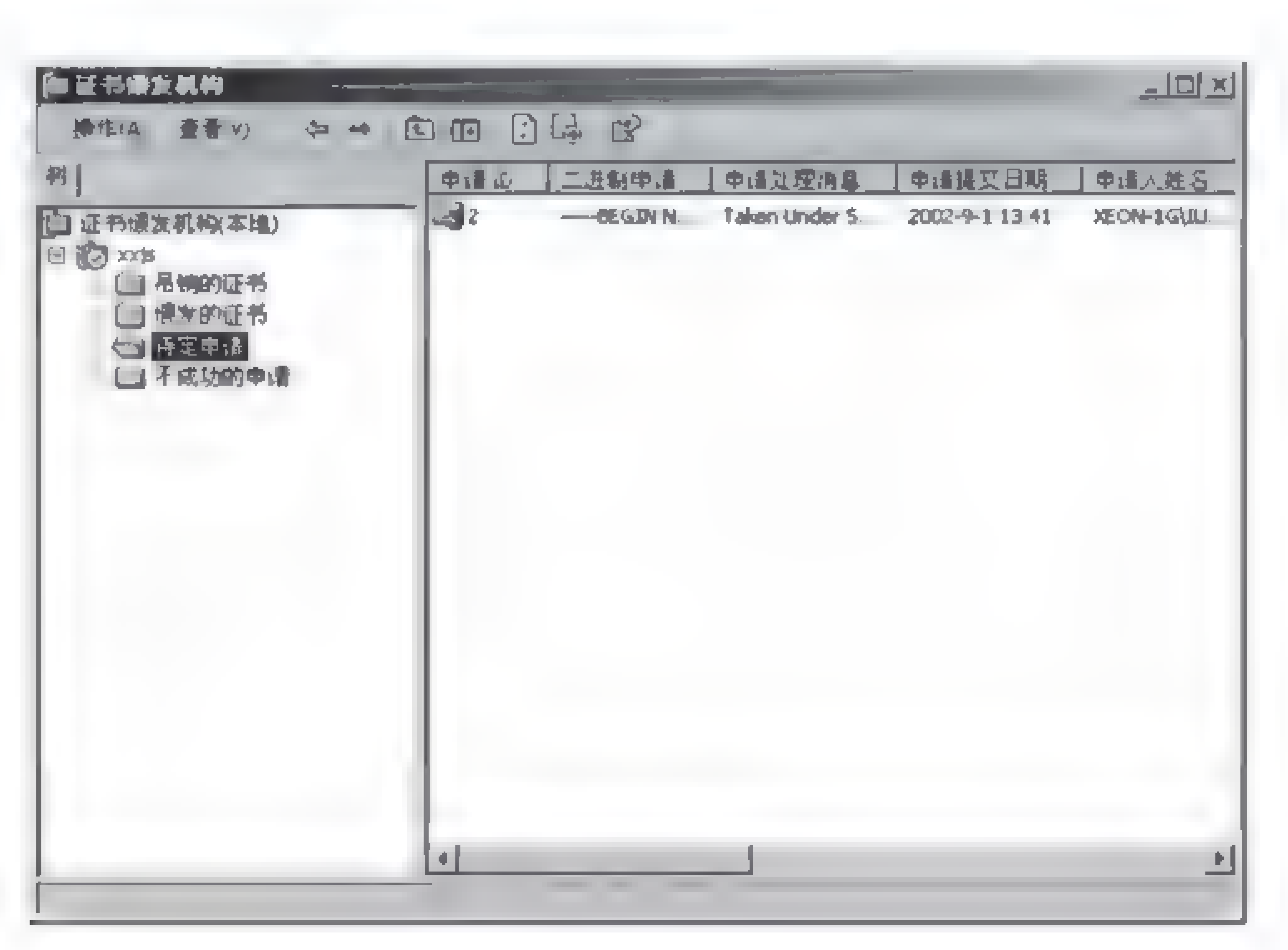


图　10-48

（4）在申请的证书处右击，选择“所有任务”选项，然后选择“颁发”选项，如图 10-49 所示。

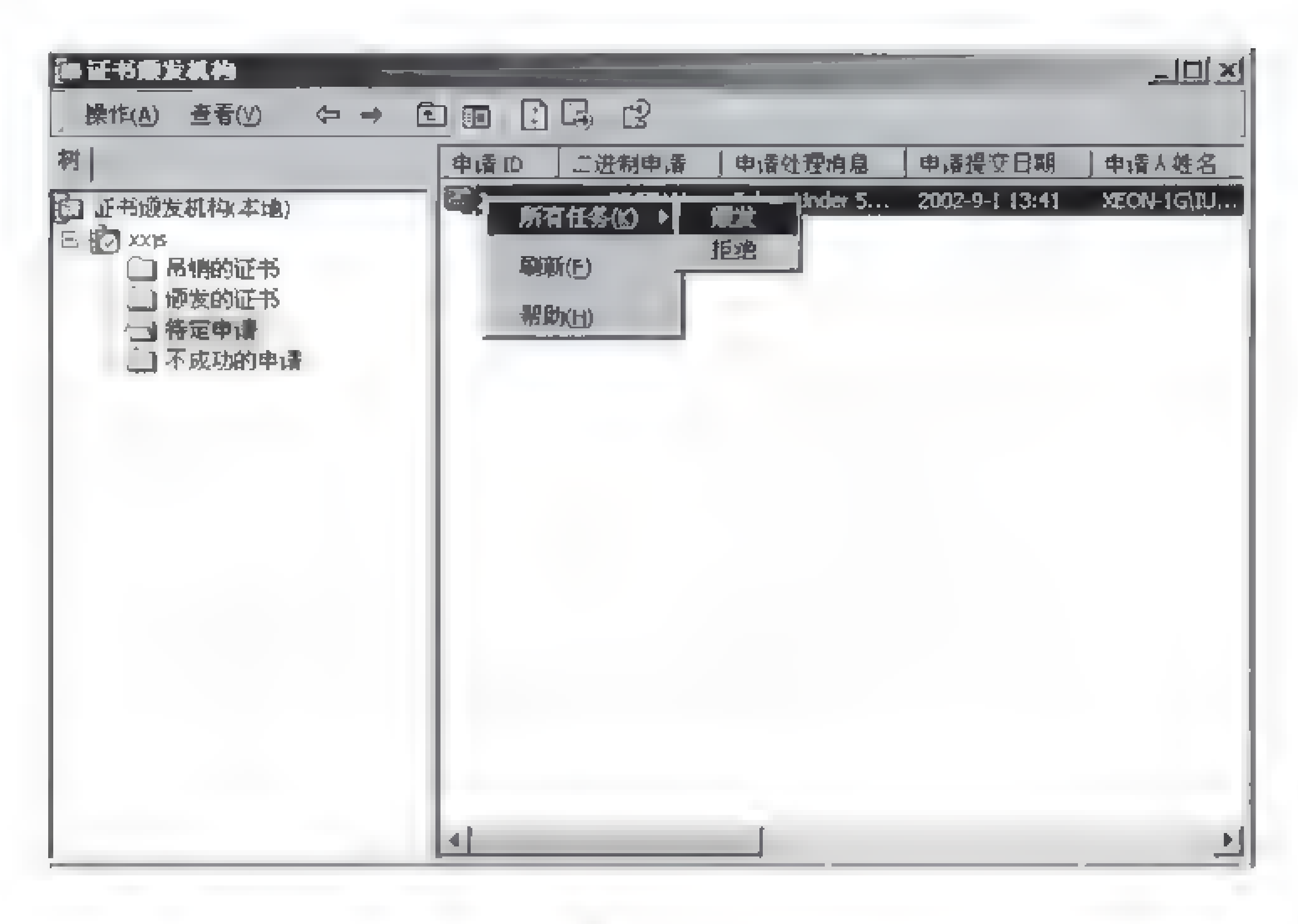


图 10-49

（5）单击“颁发证书”选项，在图 10-50 中可以看到，证书请求已经被批准颁发了。

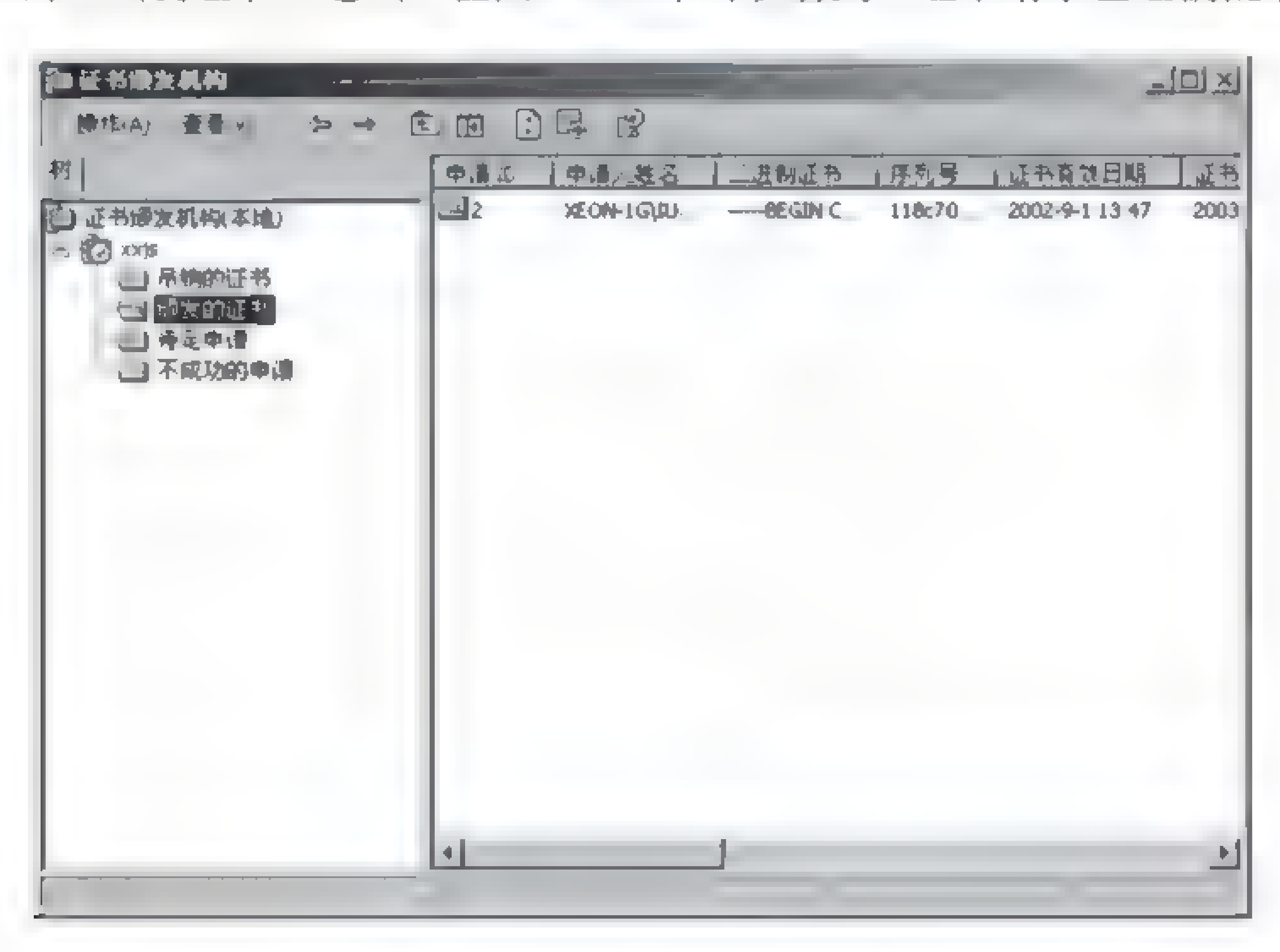


图 10-50

5. 安装服务器证书

接下来将在安装了证书服务的计算机上安装服务器证书。

（1）在 IE 的“地址”文本框中输入 http://localhost/certsrv/，打开图 10-51 所示“欢迎”对话框，选择第三项“检查挂起的证书”单选按钮。

（2）单击“下一步”按钮，出现图 10-52 所示的“检查挂起的证书申请”对话框。

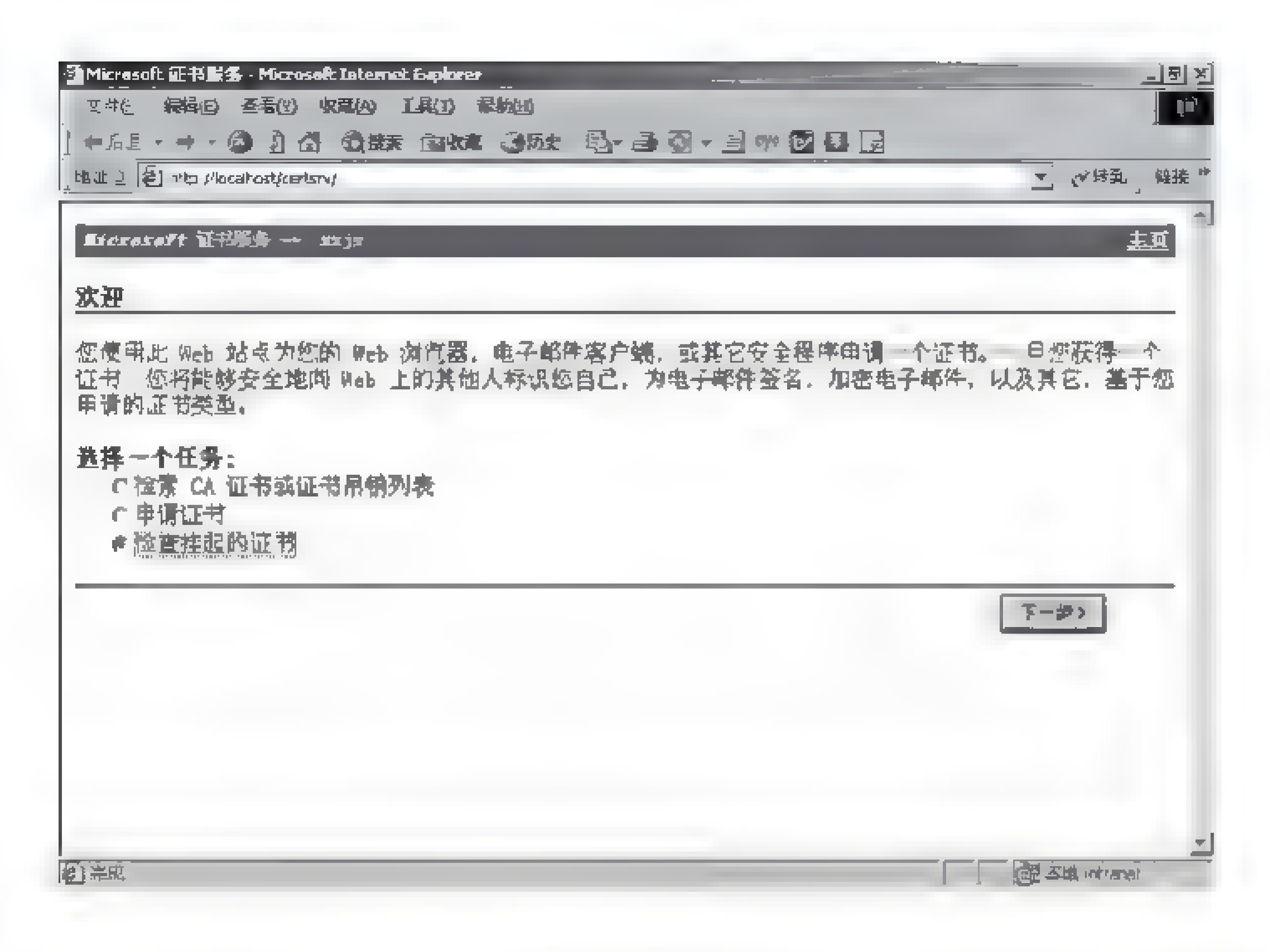


图　10-51

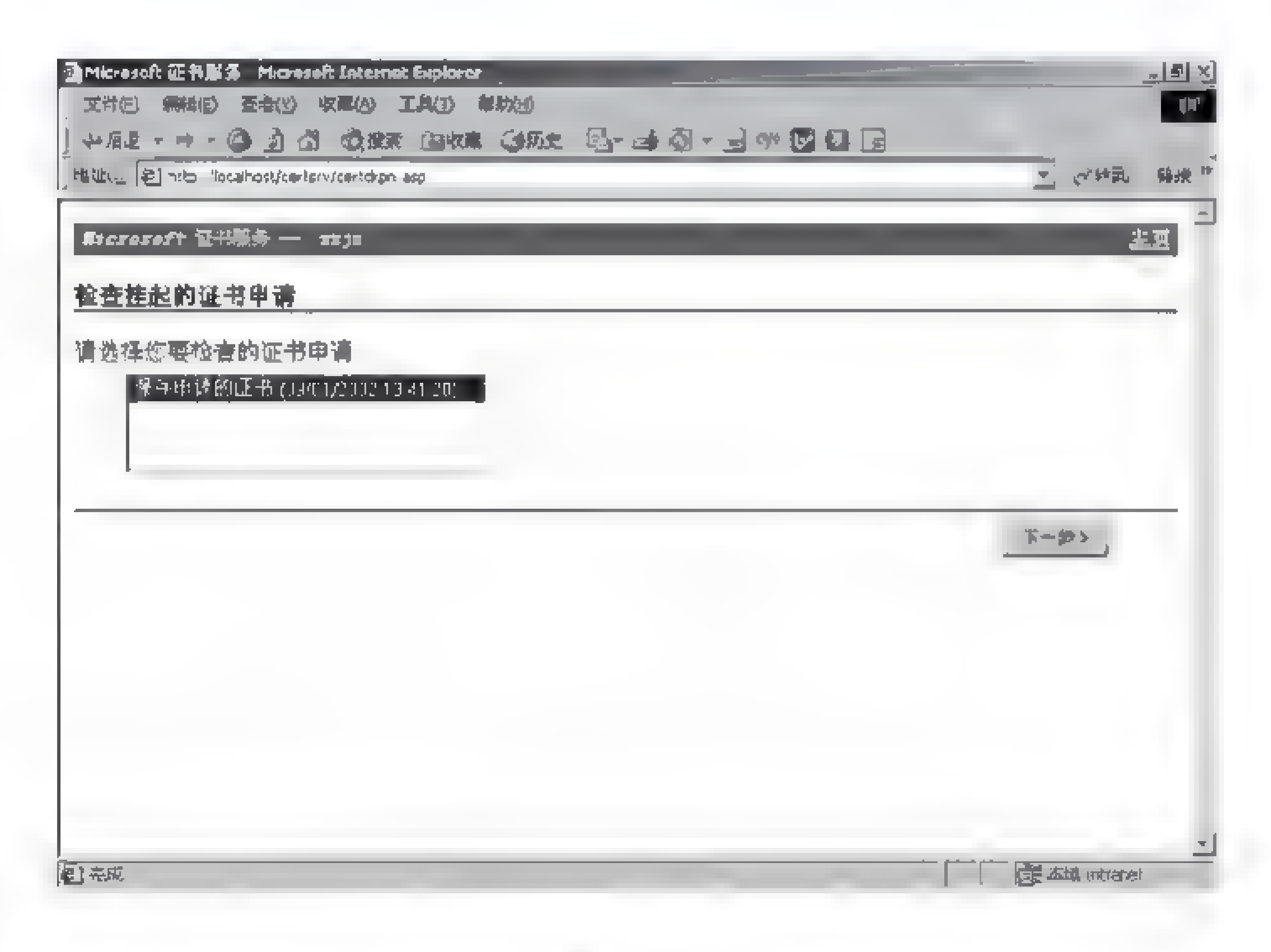


图　10-52

（3）单击“下一步”按钮，出现图 10-53 所示的“证书已发布”对话框，选择“Base64 编码”单选按钮，并单击“下载 CA 证书”链接按钮，出现“文件下载”对话框，如图 10-54 所示。

（4）单击“确定”按钮，出现图 10-55 所示的“另存为”对话框，把 certnew.cer 保存在 C:\下，单击“保存”按钮。

（5）下面开始安装服务器证书。打开 Internet 服务管理器，在“默认 Web 站点”处右击，选择“属性”选项，然后单击“目录安全性”选项卡，如图 10-56 所示。

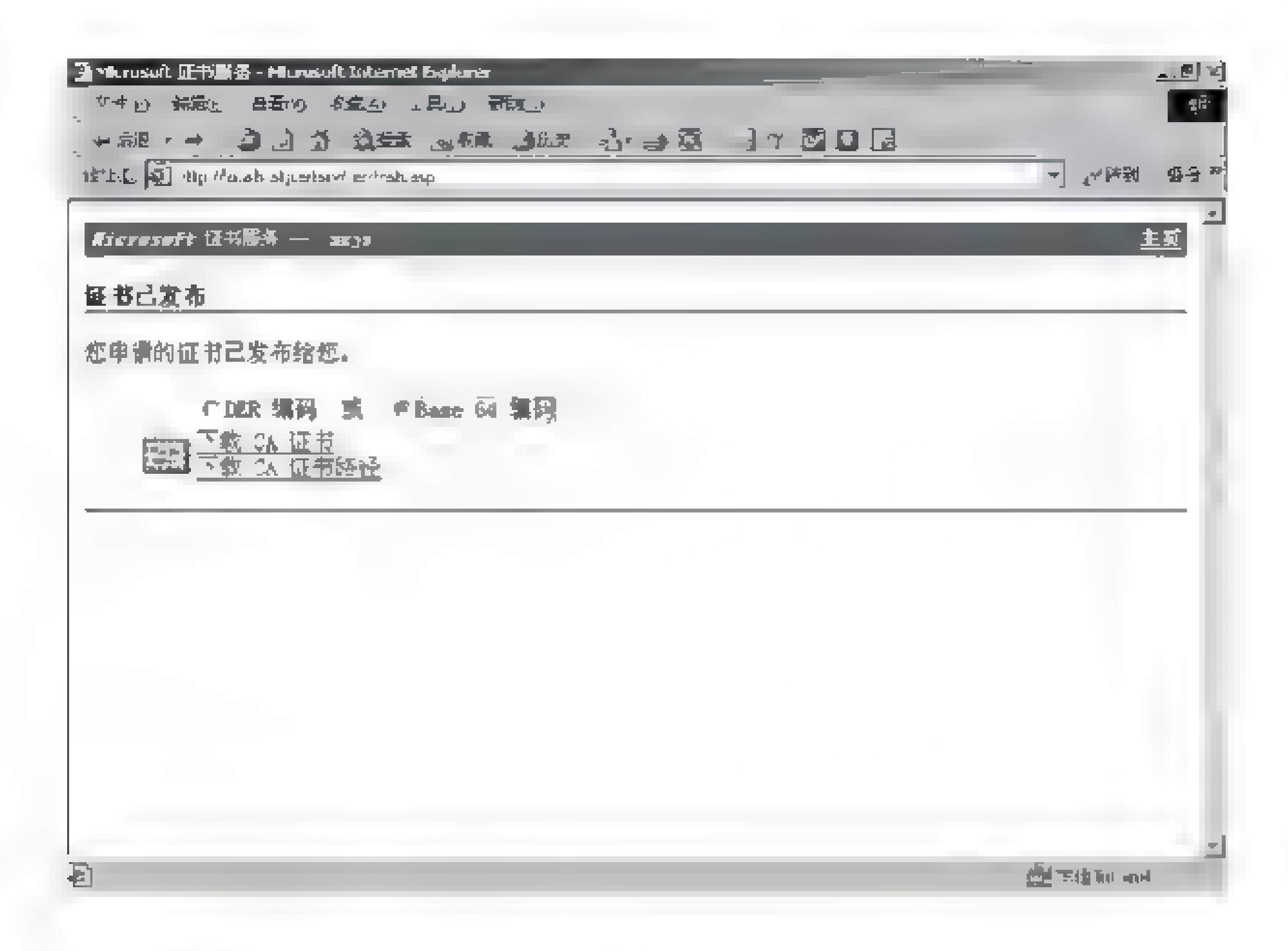


图 10-53

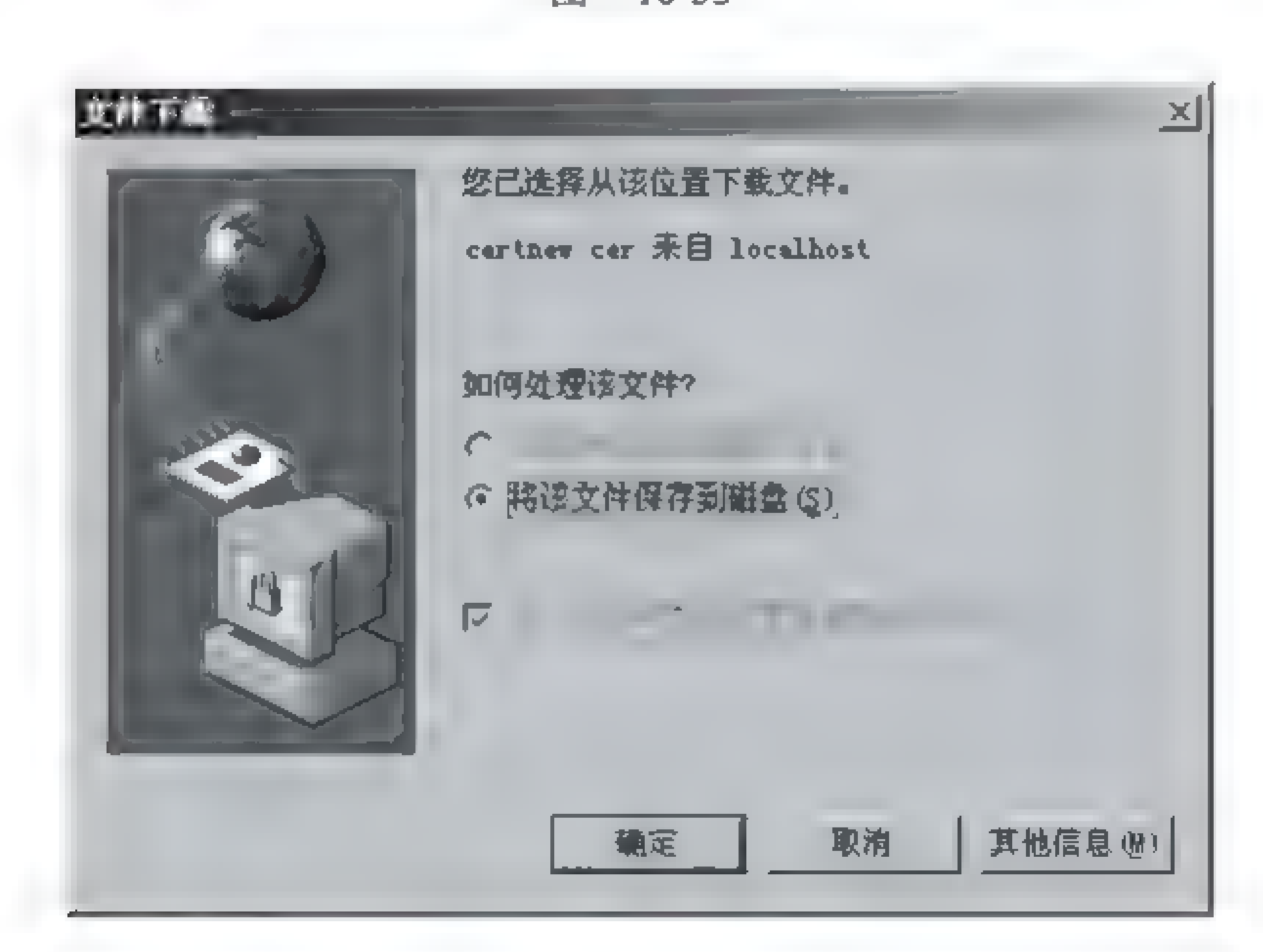


图 10-54

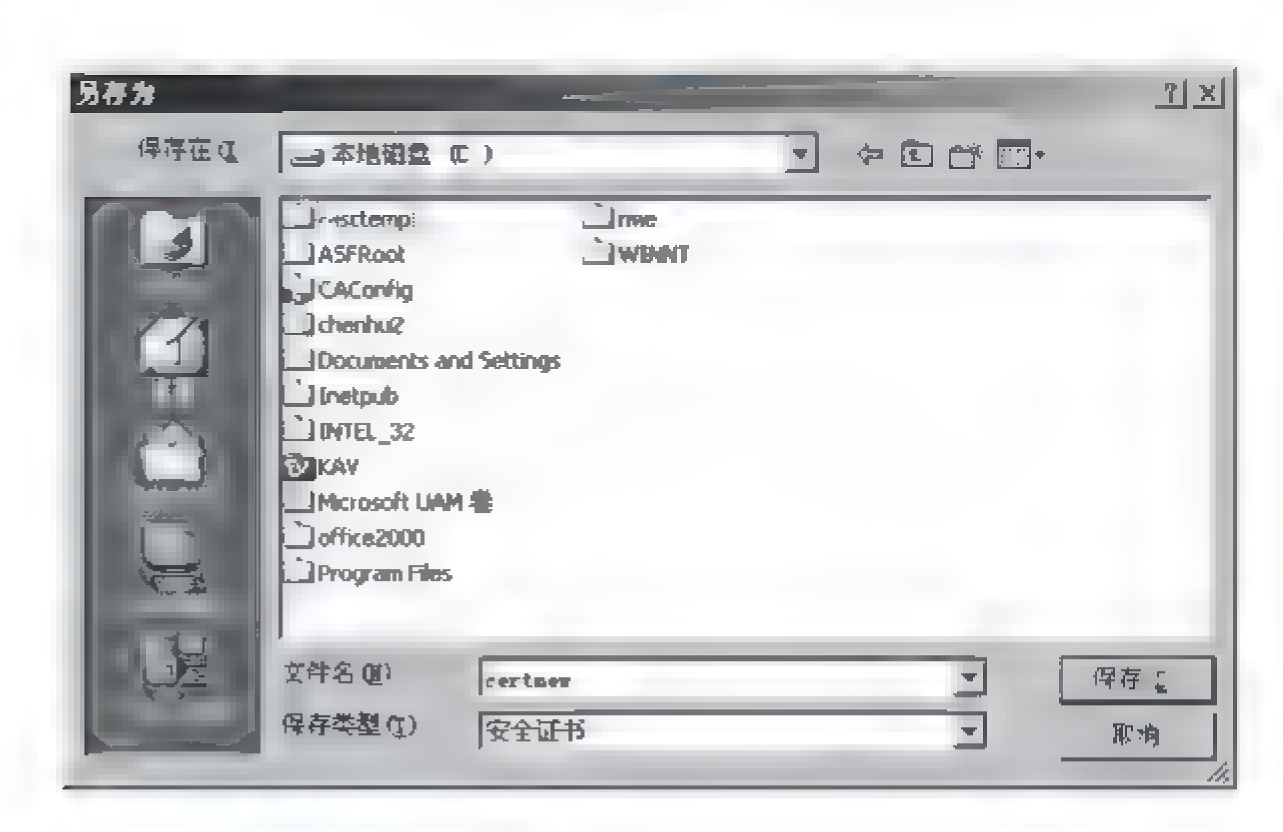


图 10-55

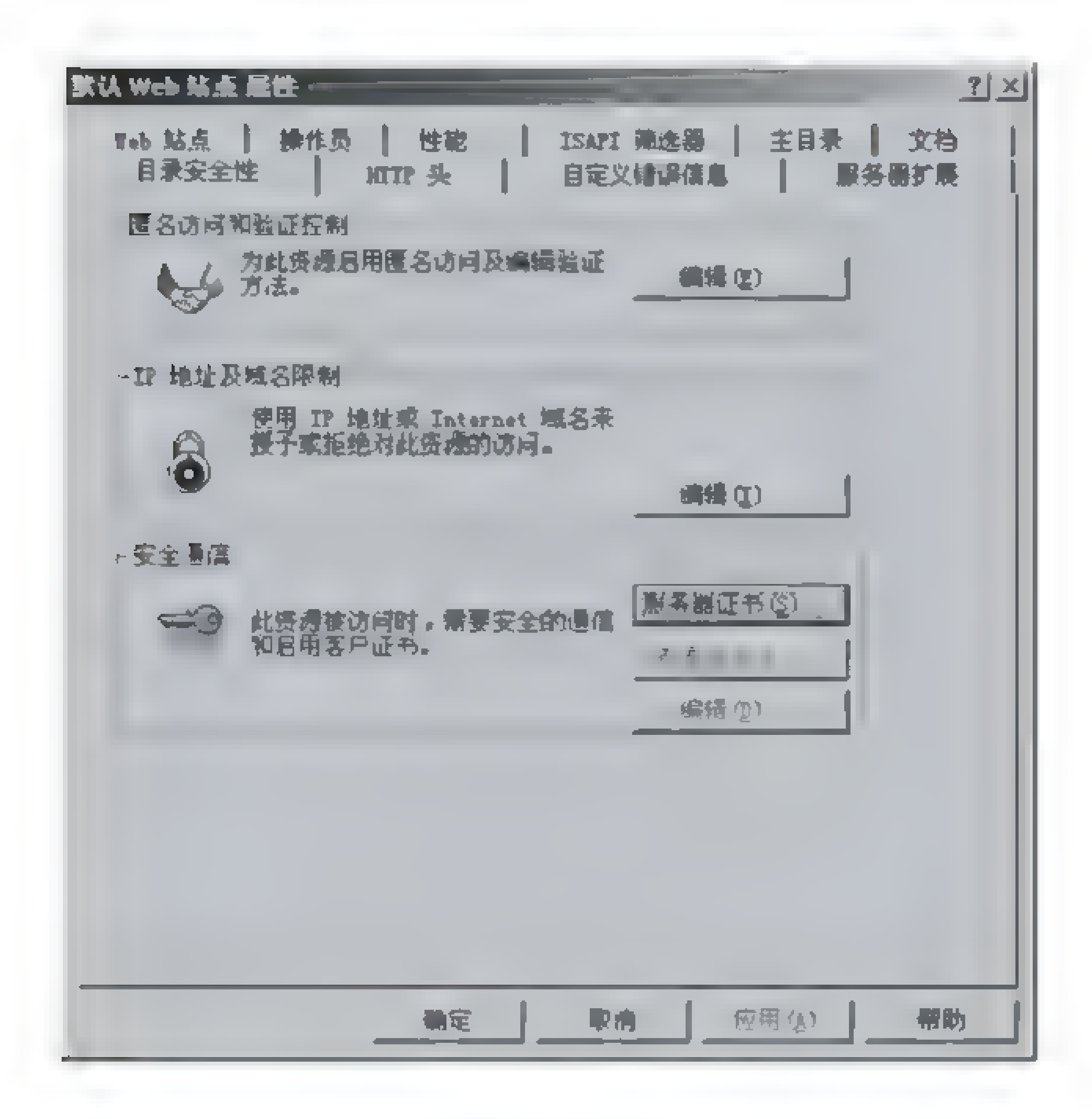


图　10-56

（6）单击“服务器证书”按钮，出现图 10-57 所示的“欢迎使用 Web 服务器证书向导”对话框。



图　10-57

（7）单击“下一步”按钮，出现图 10-58 所示的“处理挂起的请求”对话框。

（8）单击“下一步”按钮，出现“证书摘要”对话框，如图 10-59 所示。

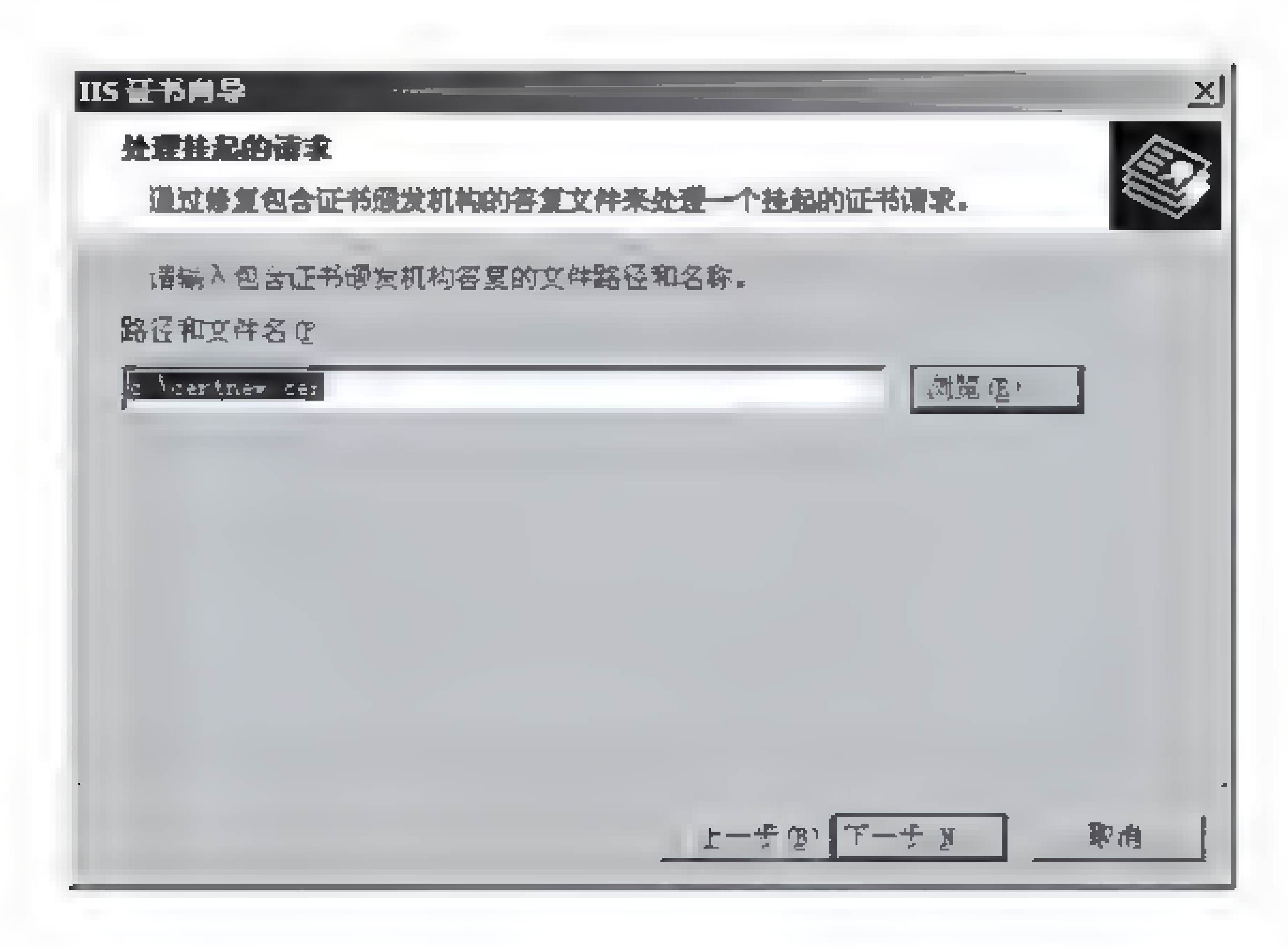


图 10-58

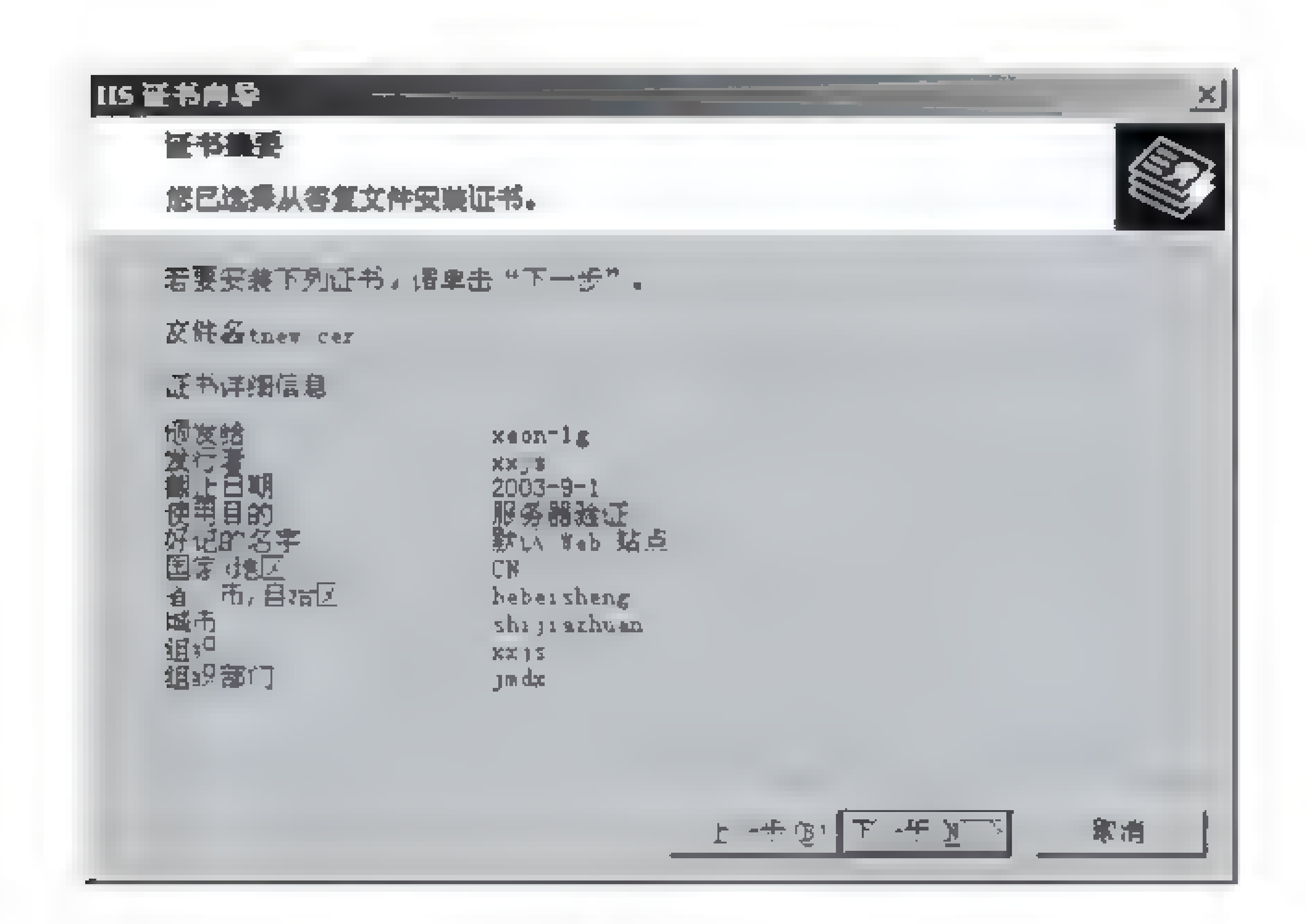


图 10-59

（9）单击“下一步”按钮，出现“证书注册”提示对话框，如图 10-60 所示。

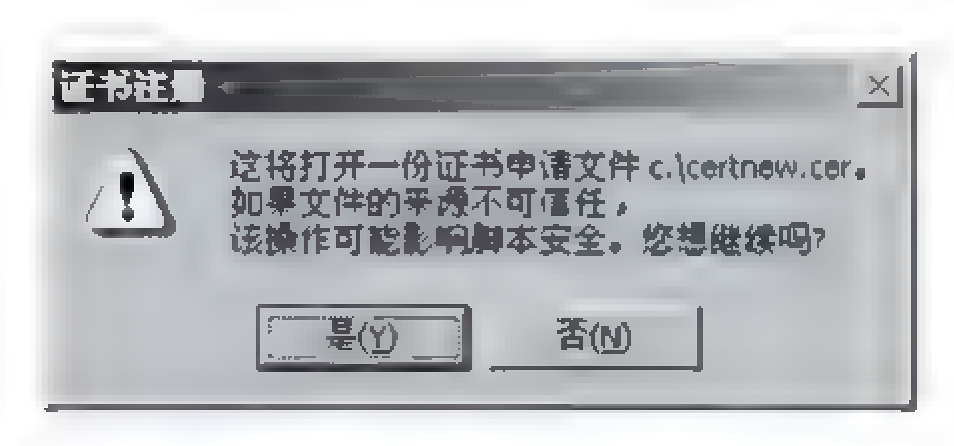


图 10-60

（10）单击“是”按钮，出现“完成 Web 服务器证书向导”对话框，单击“完成”按钮，如图 10-61 所示完成了证书的安装。



图　10-61

6. 在站点上允许使用 SSL

（1）打开 Internet 服务管理器，在“默认站点”上右击，选择“属性”选项，出现图 10-62 所示的“默认 Web 站点属性”对话框，SSL 端口号默认分配为 443。

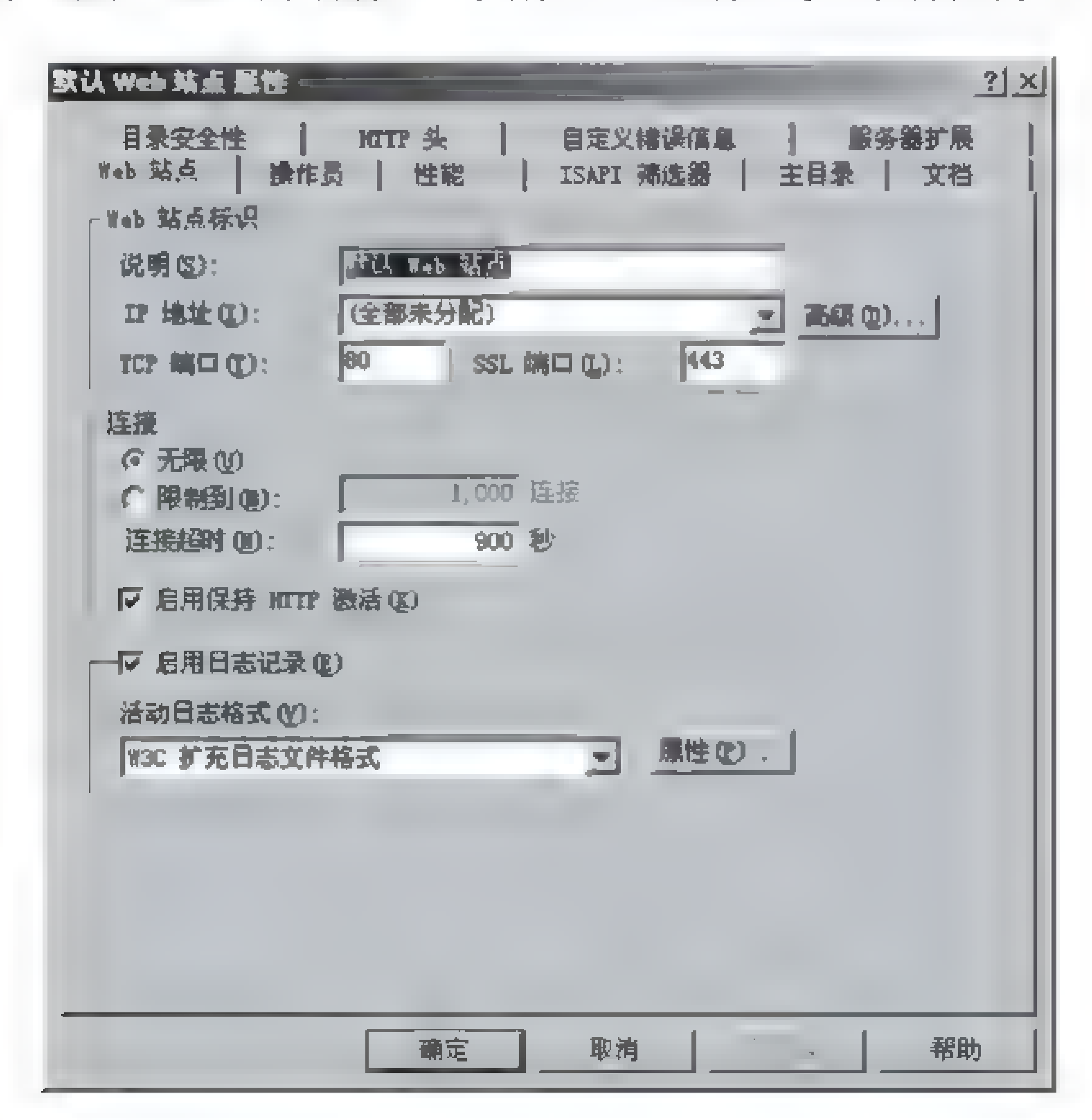


图　10-62

（2）单击“目录安全性”选项卡，如图 10-63 所示。

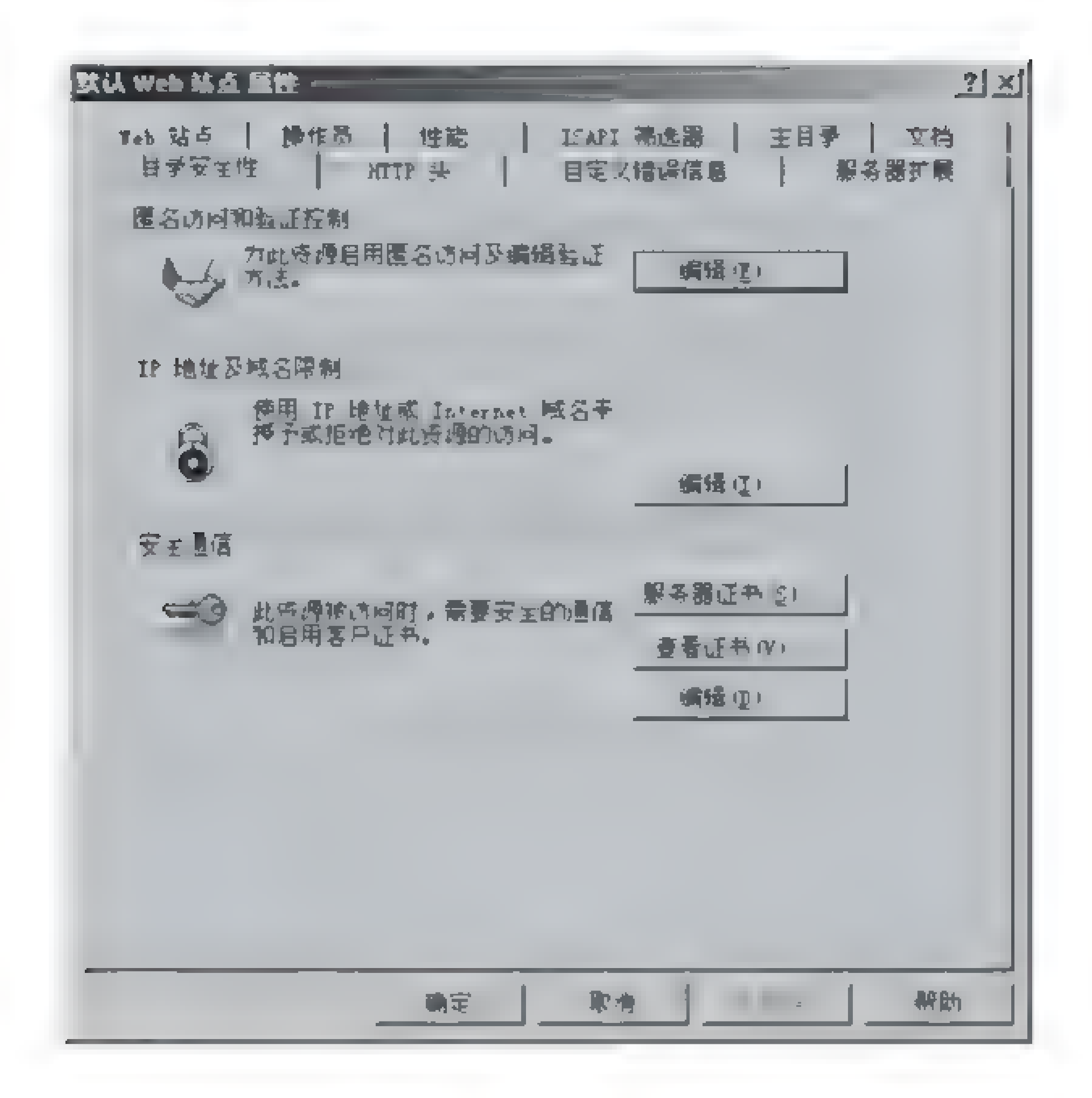


图 10-63

（3）单击“安全通信”选项区域中的“编辑”按钮，出现图 10-64 所示的“安全通信”对话框。

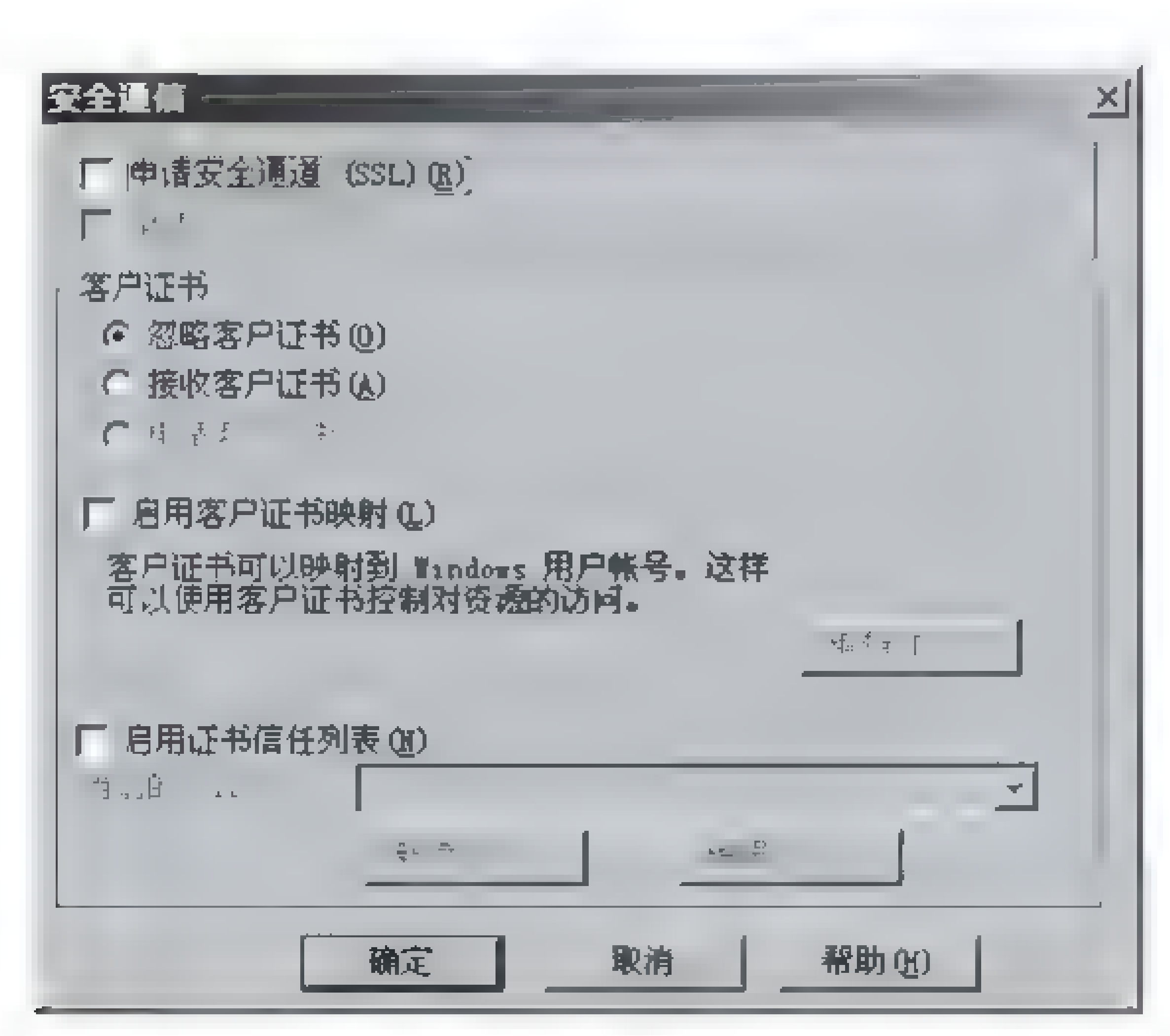


图 10-64

（4）在“安全通信”对话框中，选中“申请安全通道（SSL）”复选框。同时也可以选择申请 128 位加密。另外的“客户证书”选项和“启用客户证书映射”选项可以按自身的要求进行选择，这里采用默认值，如图 10-65 所示。

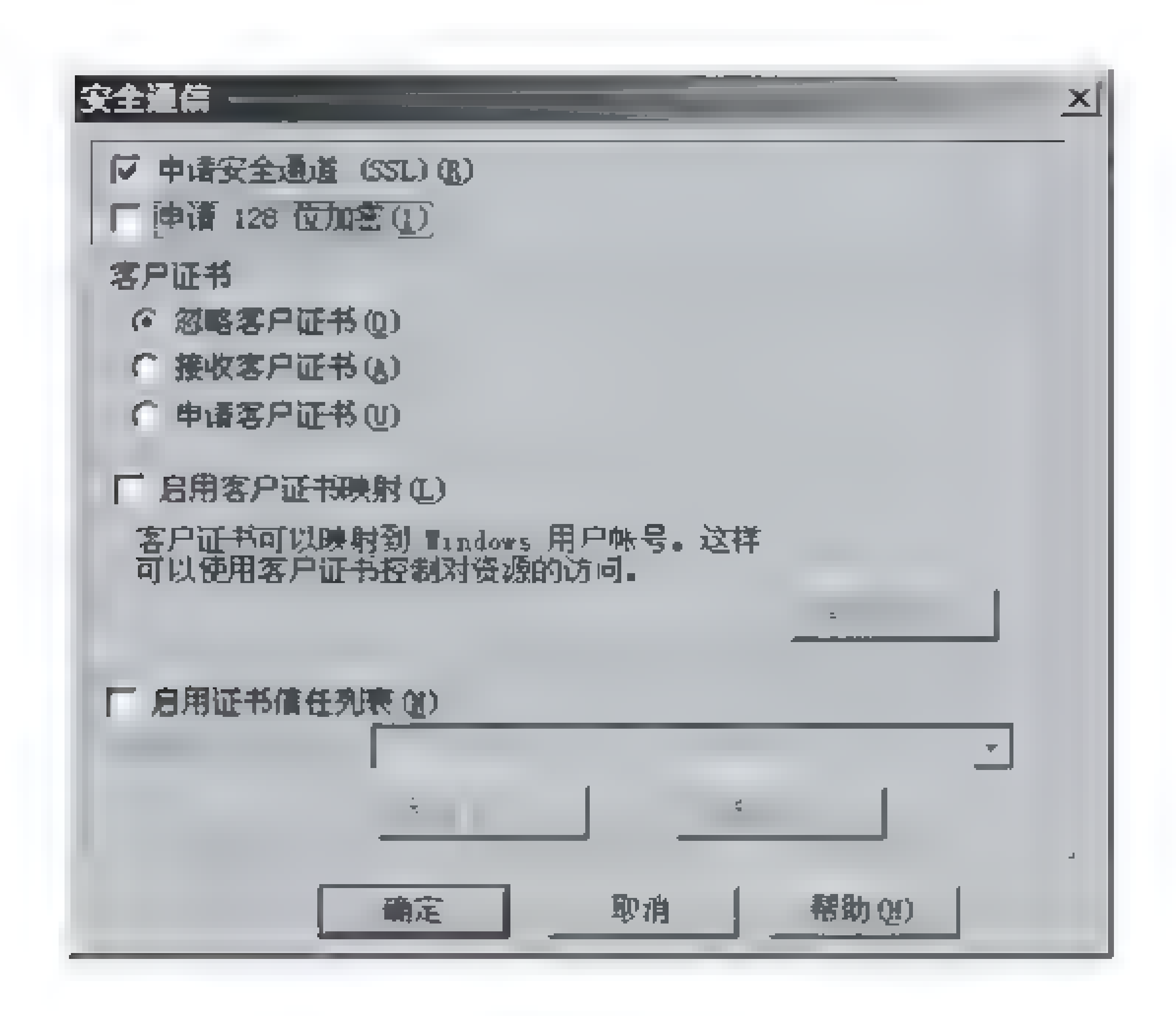


图　10-65

（5）单击“确定”按钮，返回图 10-66 所示的对话框，再单击“确定”按钮即可。

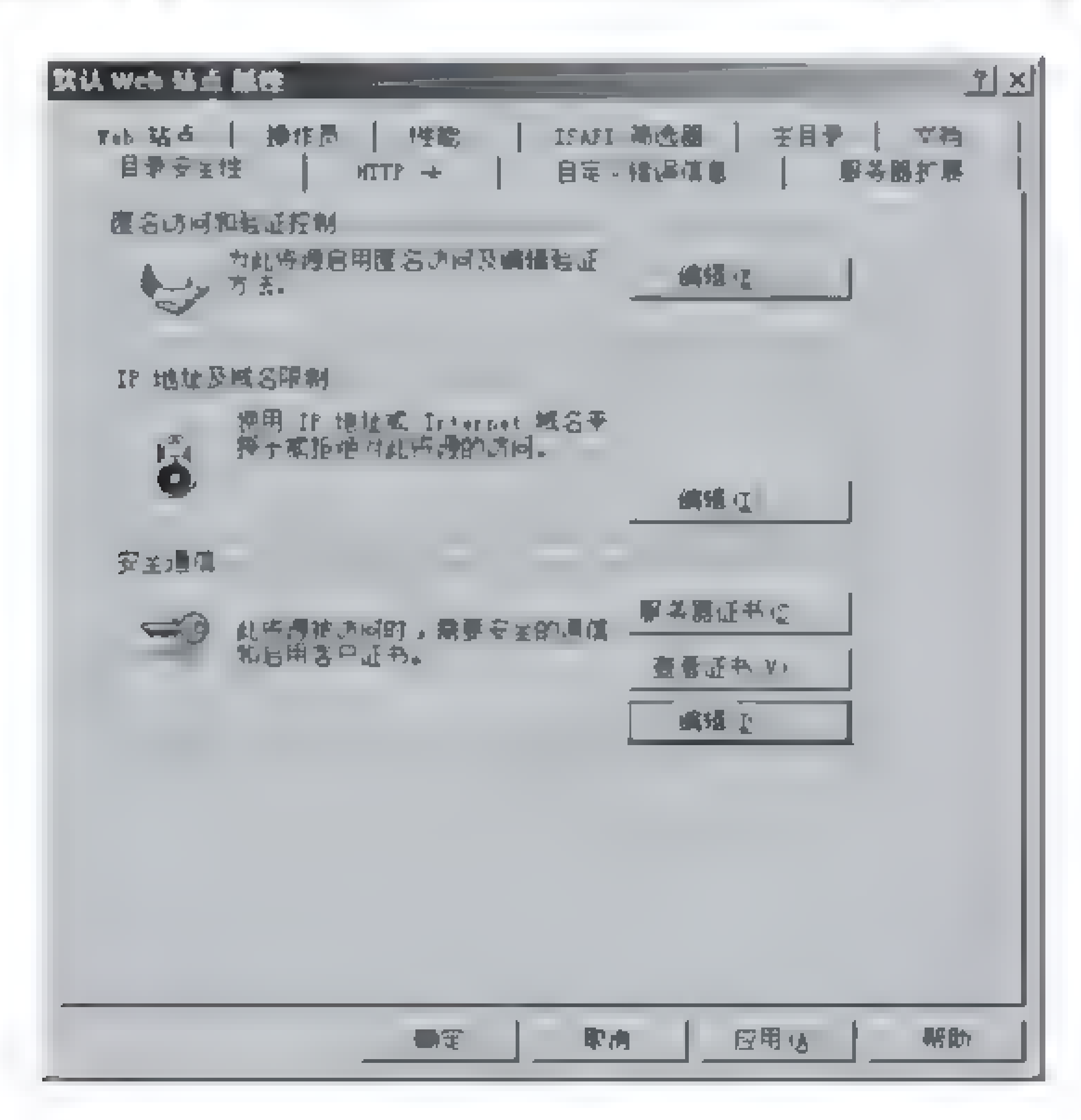


图　10-66

7. 向客户浏览器的 Root 仓库中增加 CA 证书

在浏览器和 Web 站点之间开始 SSL 通信之前，客户端必须能够认出服务器的证书是合法的。客户端必须和服务器的证书授权机构取得联系。我们的客户端是 IE 5.0。

（1）在 IE 的“地址”文本框中输入 Web 站点的地址，出现图 10-67 所示的“安全警报”提示对话框，确定以后是否要显示该警告。

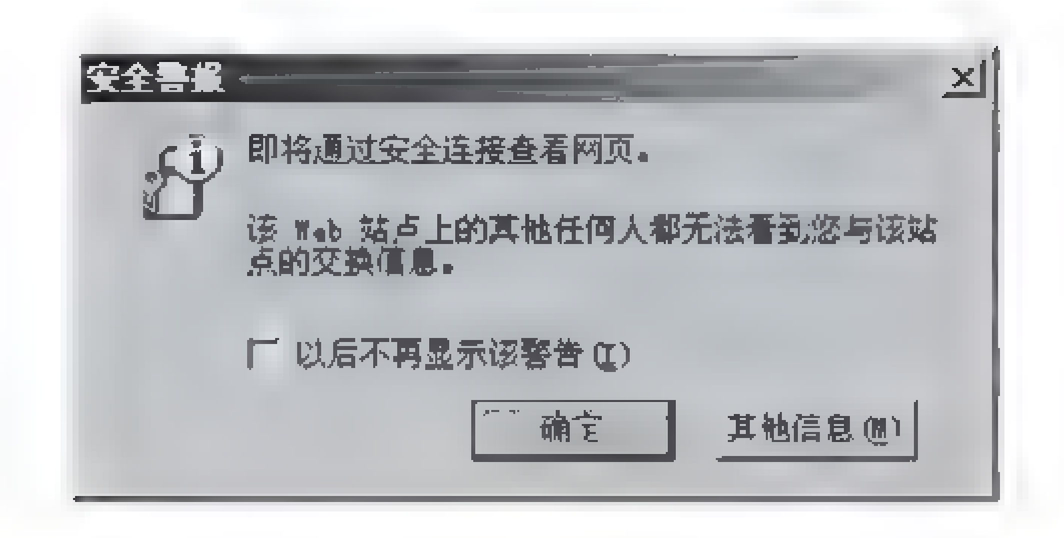


图 10-67

（2）单击“确定”按钮，出现“安全警报”对话框，如图 10-68 所示。

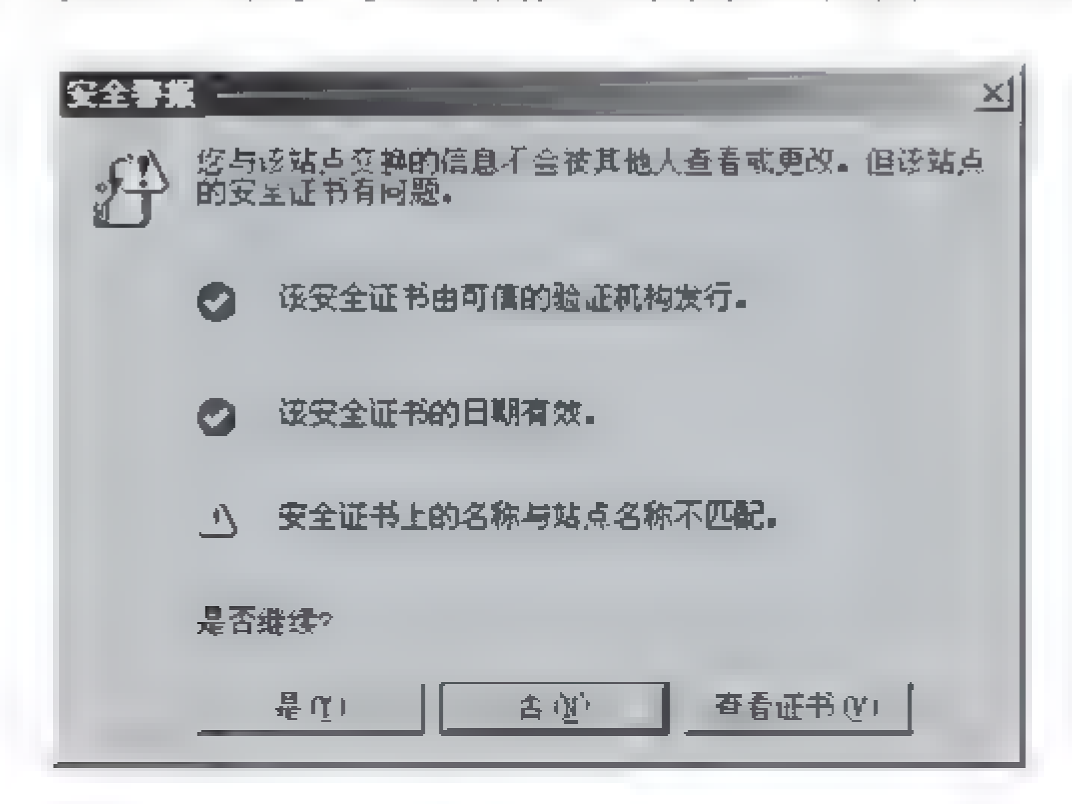


图 10-68

（3）单击“查看证书”按钮，出现图 10-69 所示的“证书”对话框。

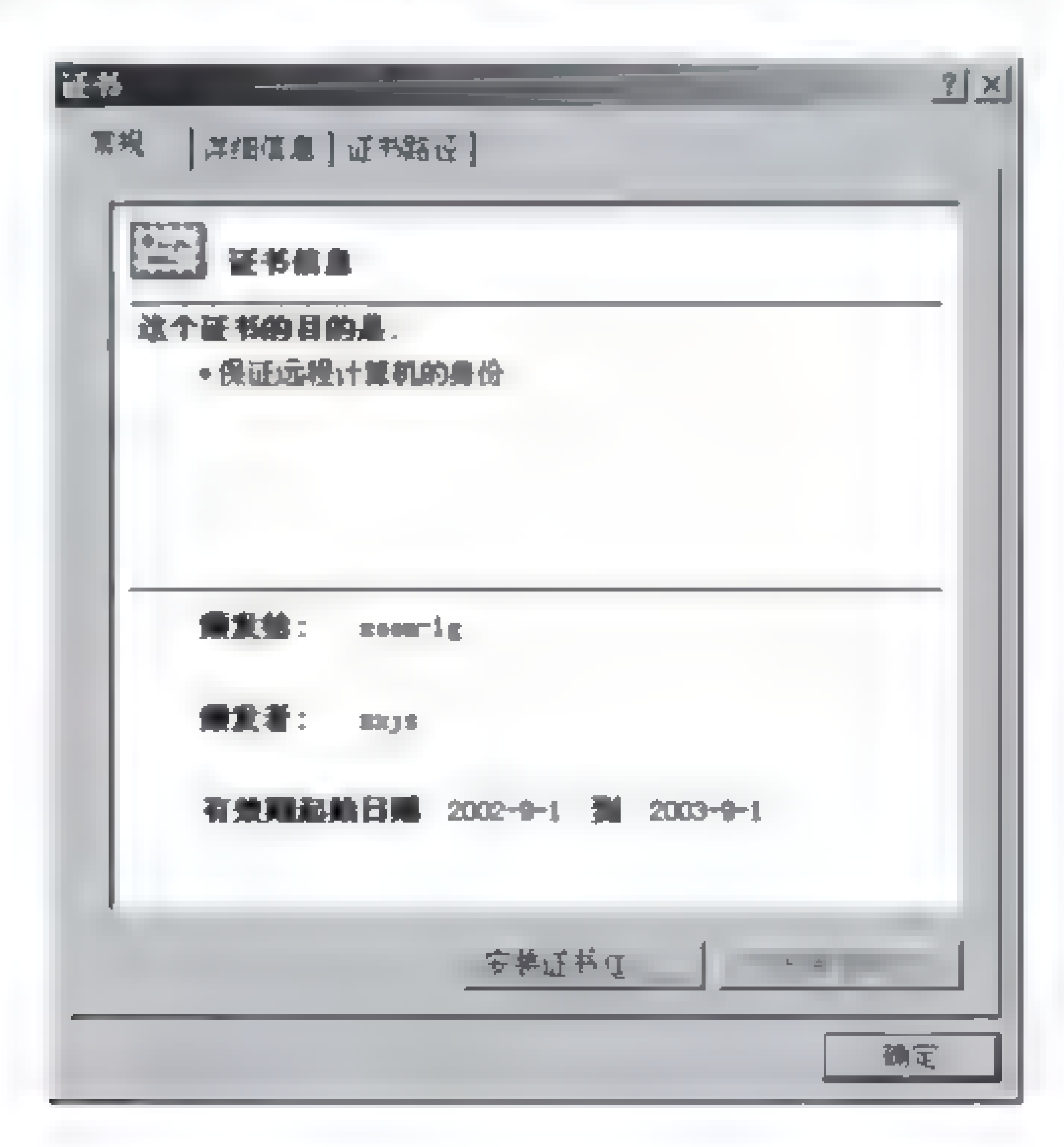


图 10-69

（4）单击“安装证书”按钮，出现图 10-70 所示的“欢迎使用证书导入向导”对话框。



图　10-70

（5）单击“下一步”按钮，出现图 10-71 所示的证书导入向导的“证书存储”对话框。

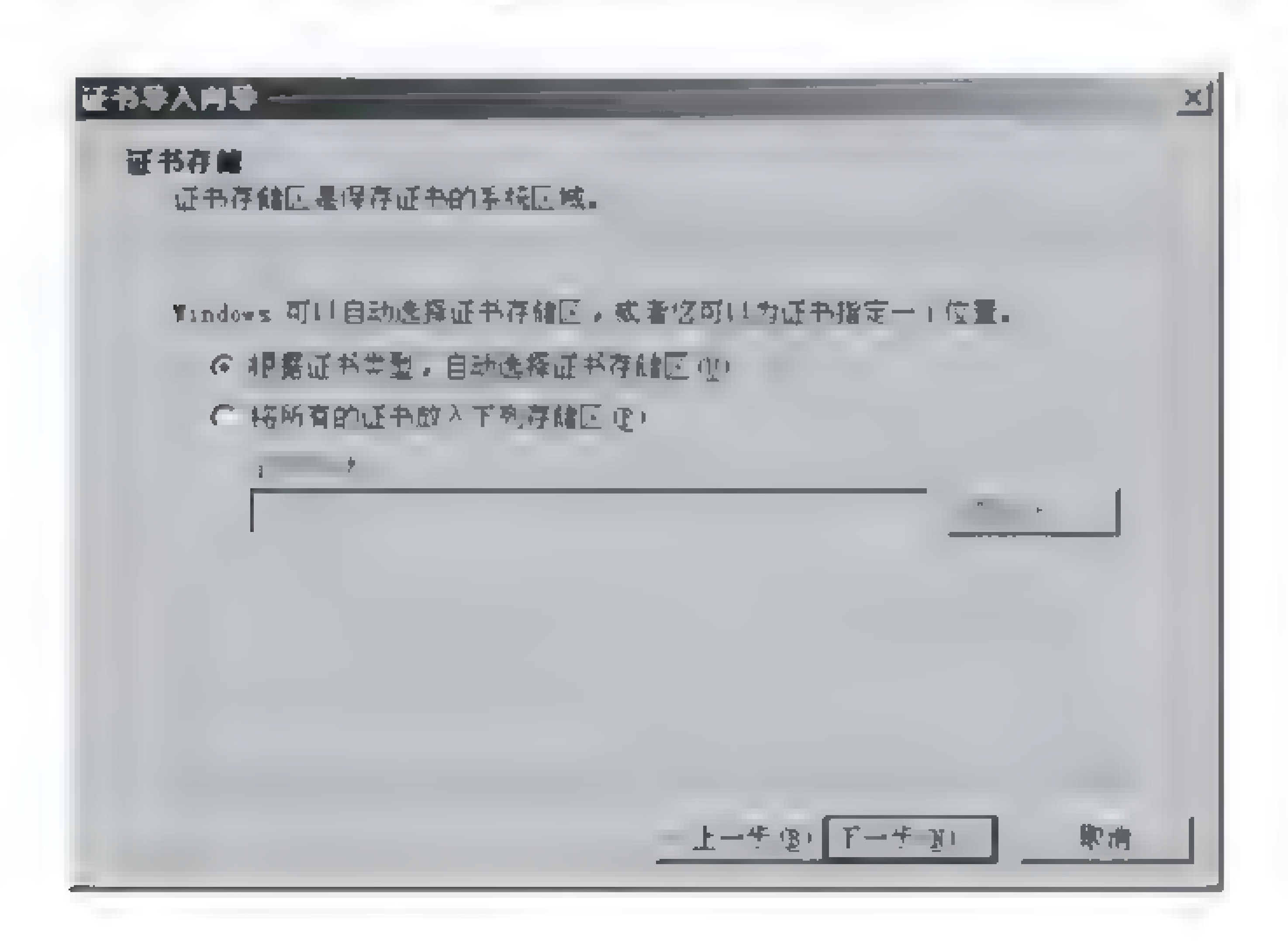


图　10-71

（6）单击“下一步”按钮，出现“正在完成证书导入向导”对话框，单击“完成”按钮，如图 10-72 所示，完成 CA 证书安装。

（7）返回图 10-68 所示的对话框，单击“是”按钮，将以安全通信的方式打开此站点。进入这个安全网页，在 IE 的状态栏里应该有一个小锁，双击这个小锁就能看到你的站点证书信息，同时也能看到整个证书链。



图 10-72

注意：可以将自己建立的网站复制到某一虚拟目录中，然后使自己的站点成为一个安全站点。

读者意见反馈

亲爱的读者：

感谢您一直以来对清华版计算机教材的支持和爱护。为了今后为您提供更优秀的教材，请您抽出宝贵的时间来填写下面的意见反馈表，以便我们更好地对本教材做进一步改进。同时如果您在使用本教材的过程中遇到了什么问题，或者有什么好的建议，也请您来信告诉我们。

地址：北京市海淀区双清路学研大厦 A 座 602 室　计算机与信息分社营销室　收
邮编：100084　　　　　　　电子邮件：jsjjc@tup.tsinghua.edu.cn
电话：010-62770175-4608/4409　　邮购电话：010-62786544

教材名称：计算机网络安全管理（第 2 版）
ISBN：978-7-302-17066-2

个人资料

姓名：__________年龄：______所在院校/专业：__________

文化程度：________通信地址：________________

联系电话：________电子信箱：________________

您使用本书是作为：□指定教材　□选用教材　□辅导教材　□自学教材

您对本书封面设计的满意度：

□很满意　□满意　□一般　□不满意　改进建议__________

您对本书印刷质量的满意度：

□很满意　□满意　□一般　□不满意　改进建议__________

您对本书的总体满意度：

从语言质量角度看　□很满意　□满意　□一般　□不满意

从科技含量角度看　□很满意　□满意　□一般　□不满意

本书最令您满意的是：

□指导明确　□内容充实　□讲解详尽　□实例丰富

您认为本书在哪些地方应进行修改？（可附页）

____

____

您希望本书在哪些方面进行改进？（可附页）

____

____

电子教案支持

敬爱的教师：

为了配合本课程的教学需要，本教材配有配套的电子教案（素材），有需求的教师可以与我们联系，我们将向使用本教材进行教学的教师免费赠送电子教案（素材），希望有助于教学活动的开展。相关信息请拨打电话 010-62776969 或发送电子邮件至 jsjjc@tup.tsinghua.edu.cn 咨询，也可以到清华大学出版社主页（http://www.tup.com.cn 或 http://www.tup.tsinghua.edu.cn）上查询。

“高等学校教材·计算机应用”系列书目

书　名	作　者	ISBN 号
C++语言程序设计教程与实验	温秀梅等	9787302081432
CAD 二次开发及其工程应用	王育琨等	2007 年 12 月出版
C 程序设计案例教程	王岳斌等	9787302136798
Delphi 程序设计教程	杨长春等	9787302112136
Excel 在数据管理与分析中的应用	杜茂康	9787302104001
Internet 应用基础教程	徐详征等	9787302084945
Internet 实用技术与网页制作	孙芳等	9787302112211
Java 2 程序设计基础	陈国君等	9787302120551
Java 程序设计之网络编程	李芝兴等	9787302123224
PowerBuilder 数据库应用开发技术	卢守东	9787302127291
Protel 电路设计教程（第 2 版）	江思敏等	9787302134879
SolidWorks 及 Cosmos/Motion 机械仿真设计	张晋西	9787302140559
SPSS 统计分析实例精选	周爽	9787302124344
Visual Basic 语言程序设计教程与实验	丁学钧等	9787302105671
Visual Basic 程序设计与应用开发案例教程	曾强聪	9787302091349
Visual Basic 程序设计综合教程	朱从旭等	9787302104322
Visual C++程序设计——基础与实例分析	朱晴婷等	9787302081449
Visual FoxPro 8.0 实用教程	李明等	9787302123125
Visual FoxPro 程序设计	程学先等	9787302129967
Visual FoxPro 程序设计基础	余坚	9787302133216
Visual FoxPro 程序设计实验与学习指导	余坚	9787302133629
Visual FoxPro 数据库基础教程	姜桂洪等	9787302132509
Visual FoxPro 数据库应用教程与实验	徐辉等	9787302098560
Web 技术导论	郝兴伟	9787302101185
Windows 程序设计技术	刘腾红等	9787302095453
办公自动化概论	张锐昕等	9787302088530
操作系统教程与实验	胡明庆等	9787302137511
操作系统实验教程（Windows 版）	姚卫新	9787302102519
单片机原理、接口及应用——嵌入式系统技术基础	李群芳等	9787302101802
电子档案管理基础	王萍等	9787302124542
多媒体技术毕业设计指导与案例分析	贺雪晨等	9787302102526
多维数据分析原理与应用	姚家奕等	9787302083771
计算机辅助设计教程	张秉森等	9787302101178
计算机控制技术	姜学军	9787302107910
计算机外围设备	张钧良	9787302100881
计算机网络技术基础教程	刘四清等	9787302082057
计算机网络技术及应用教程	杨青等	9787302143338
计算机网络技术教程——基础理论与实践	胡伏湘等	9787302080732
计算机网络教程	王群	9787302120193
计算机网络实用技术教程	李冬等	9787302140108
计算机网络与通信	陈向阳等	9787302118619

书 名	作 者	ISBN 号
计算机网络与应用	石良武	9787302104926
计算机维修技术	易建勋	9787302110453
计算机信息技术应用基础	杜茂康等	9787302082392
计算机信息技术应用教程	彭宗勤等	9787302109341
计算机应用基础	刘毅等	9787302112563
计算机应用技术基础	范慧琳等	9787302132608
计算机应用技术学习指导与实验教程——例题精解与练习、上机实践	范慧琳等	9787302133155
计算机英语实用教程	张强华	9787302090731
计算机硬件技术基础	曹岳辉等	9787302119715
计算机与网络应用基础教程	朱根宜	9787302086307
建筑 CAD 技术应用教程	吴涛	9787302091929
局域网技术与应用	李琳	9787302087571
局域网与城域网技术	王文鼐等	9787302140696
科技情报检索	田质兵等	9787302089070
面向对象程序设计 Visual C++ 6.0 教程题解与实验指导	陈天华	9787302133735
面向对象程序设计与 Visual C++ 6.0 教程	陈天华	9787302123118
面向对象技术与 Visual C++	甘玲	9787302090700
面向对象技术与 Visual C++学习指导	甘玲等	9787302123231
软件技术基础教程	周肆清等	9787302116981
实用计算机技术——公安司法应用实践	汤艳君等	9787302133766
数据结构——C++语言描述	朱振元等	9787302142157
数据库及其应用	肖慎勇等	9787302140757
数据库及其应用学习与实验指导教程	肖慎勇等	9787302104728
数据库系统及应用（Visaul FoxPro）第二版	邓洪涛	9787302142966
数据库系统及应用（visual ForPro）	邓洪涛	9787302086253
数据库与网络技术	翟延富	9787302124962
数据通信与网络应用	吴金龙等	9787302128649
统计分析方法——SAS 实例精选	周爽	9787302091295
图形图像处理应用教程	张思民等	9787302124795
网络工程规划与设计	陈向阳等	9787302143086
网络基础与应用实务教程	段宁华	9787302124300
网络医学信息应用	刘汉义等	9787302142690
网络远程教学技术基础（含上机指导）	黄景碧等	9787302115595
网络远程教学资源设计开发（化学）	黄景碧等	9787302150848
网页设计教程	侯文彬等	9787302091875
网站建设——基于 Windows Server 2003 和 Linux 9	葛秀慧	9787302101819
微机组装与维护	查志琴等	9787302103417
信息检索	陈雅芝	9787302120513
运筹学算法与编程实践——Delphi 实现	刘建永等	9787302093619
中文信息处理技术——原理与应用	李宝安等	9787302112006